fare elettronica

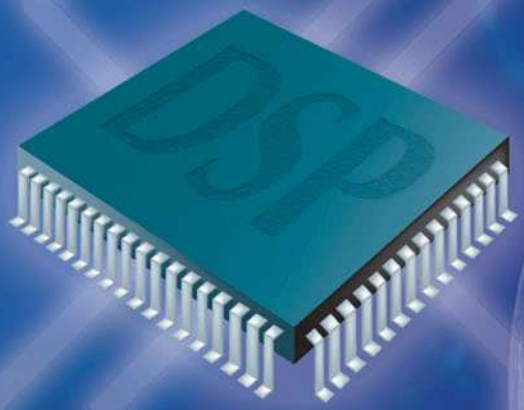
CULTURA ELETTRONICA APPLICATA

130 pagine!

DICEMBRE
n° 246 • Anno 21

CPLD By Example
Sistema hardware di sviluppo

Introduzione ai Processori DSP



MIKROBASIC
Interfaccia utente

LA TECNICA DDS
Generatore di segnali a frequenza variabile

USARE PROTEUS
ARES - Tecniche di base per il piazzamento e lo sbroglio di PCB

ALIMENTATORI SWITCHING
Analisi di uno switching off-line (parte II)

VITAMINA C
Sistemi operativi

Biofeedback
Trasmissione a distanza

ELETTRONICANDO
Circuiti reazionati, oscillatori e reti di sfasamento

INWARE EDIZIONI
ISSN 1591-2272
9 771591 227008 50246

€ 5,50

Poste Italiane Spa - Spedizione in abbonamento Postale - D.L. 353/2003 (conv. In L. 27/02/2004 n. 46) art. 1, comma1, DCB Milano. In caso di mancato recapito, restituire all'editore che si impegna a pagare la relativa tassa presso il CPM di Roserio - Milano

Una nuova era

Sfogliavo giusto l'altra sera una delle riviste di inizio anno scorso, ove concludevo il mio intervento introduttivo con una frase profetica "non dormiremo sugli allori e miglioreremo la rivista sempre di più".
Chissà quanti di voi hanno notato come i recenti numeri di Fare Elettronica, nel nuovo formato A4, non entrano più nelle librerie standard. Abbiamo ricercato il nuovo formato per distinguere la nuova rivista rispetto alle precedenti, per segnare in maniera ancora più marcata l'inizio della *nuova era*. Proprio così, i nostri lettori ed i sempre più numerosi abbonati, potranno godere di pubblicazioni (ribadisco: pubblicazioni) sempre più in linea con i tempi moderni, con le esigenze di oggi. I CD-ROM, i libri e i numeri speciali che stiamo preparando sono infatti iniziative nate per soddisfare a 360 gradi le esigenze del nostro pubblico. Soprattutto gli speciali rappresentano l'iniziativa editoriale che ci vedrà maggiormente impegnati l'anno prossimo. Fare Elettronica incrementerà la sua offerta nel tentativo di esaudire ogni vostro desiderio, sia esso di natura teorica che pratica, nei diversi campi: firmware, elettronica analogica, digitale, componentistica attiva e passiva.
Siamo lieti di constatare un caldo clima di fiducia intorno a noi in un periodo in cui, per l'editoria tecnica specializzata, i ritmi non sono solitamente così frenetici. Essere in controtendenza è stimolante, ma necessita un'attenzione particolare, poiché commettere errori è molto più facile. Per questo motivo sapremo fare tesoro di tutti i commenti che vorrete farci pervenire, siano essi lodi o critiche. La sfida è importante e noi l'abbiamo accettata da lontano, ma adesso è davvero cominciata la *nuova era* e anche tu, caro lettore, stai contribuendo alla realizzazione di un progetto di cui proprio tu sarai il principale beneficiario.

Questo mese parliamo di *DSP*, dispositivi ormai divenuti il cuore del mondo multimediale che ci circonda. È proprio grazie a questi potenti dispositivi che otteniamo foto digitali ad alta qualità o che godiamo la visione di un DVD ad alta risoluzione con audio multi canale. Questo articolo introduttivo vuole portarvi a capirne un po' di più, a comprenderne il funzionamento.
Giungono alla conclusione, dopo averci accompagnato per molti mesi, i corsi *Vitamina C* ed *Alimentatori Switching*, voglio quindi ringraziare *Romano Bernarducci* e *Antonio Di Stefano* per l'ottimo lavoro svolto.
Continuano invece le serie *Mikrobasic, Elettronicando, Usare Proteus* ed *Amplificatori Operazionali dalla A alla Z*. Sul fronte dei progetti, vi proponiamo una versione migliorata del monitor cardiaco portatile presentato qualche mese fa, il quale, oltre ad offrire una maggiore precisione, trasmette il segnale a distanza. Completa il numero un nuovo progetto dedicato alla *tecnica DDS: generatore di segnale a frequenza variabile*.

Concludo questo mio intervento ricordandovi che a Gennaio sarà disponibile in tutte le edicole - **Firmware** - un nuovo bellissimo numero speciale dedicato ai microcontrollori PICmicro.
Se sei abbonato a Fare Elettronica richiedi subito la tua copia telefonando allo *02-66504794* oppure su *www.farelettronica.com/firmware*, ti sarà recapitata a soli €3,00 anzichè €6,00. Se non sei abbonato non perdere questa fantastica occasione, abbonati subito!
Non mi resta che farvi i nostri migliori auguri per uno splendido Natale ed un brillante nuovo anno. Ci ritroveremo in edicola a Gennaio 2006.

Tiziano Galizia
t.galizia@farelettronica.com

fare elettronica
CULTURA ELETTRONICA APPLICATA

www.farelettronica.com

DIRETTORE RESPONSABILE
Antonio Cirella

DIRETTORE DI REDAZIONE
Tiziano Galizia

DIRETTORE TECNICO
Maurizio Del Corso

HANNO COLLABORATO IN QUESTO NUMERO
Giovanni Di Maria, Riccardo Nicoletti, Vittorio Marradi, Salvatore Torrisi, Enrico Raffone, Antonio Di Stefano, Romano Bernarducci, Nico Grilloni, Agostino Rolando.

DIREZIONE • REDAZIONE • PUBBLICITÁ
INWARE srl - Via Cadorna, 27/31 - 20032 Cormano (MI)
Tel. 02.66504794 - 02.66504755 - Fax 02.66508225
info@inware.it - www.inwaredizioni.it
Redazione: redazione@farelettronica.com

PROGETTO GRAFICO E IMPAGINAZIONE
Graficonsult - Milano

STAMPA
ROTO 2000 - Via L. da Vinci, 18/20 - 20080, Casarile (MI)

DISTRIBUZIONE
Parrini & C. S.p.a. - Viale Forlanini, 23 - 20134, Milano

UFFICIO ABBONAMENTI
INWARE srl - Via Cadorna, 27/31 - 20032 Cormano (MI)
Per informazioni, sottoscrizione o rinnovo dell'abbonamento:
abbonamenti@farelettronica.com
Tel. 02.66504794 - Fax. 02.66508225
L'ufficio abbonamenti è disponibile telefonicamente dal lunedì al venerdì dalle 14,30 alle 17,30

Poste Italiane S.p.a. - Spedizione in abbonamento Postale
D.L. 353/2003 (conv. In L. 27/02/2004 n. 46) art. 1, comma1, DCB Milano.
Abbonamento per l'Italia: € 45,00
Abbonamento per l'estero: € 115,00

Gli arretrati potranno essere richiesti, per iscritto, al seguente costo:
Numero singolo: € 7,50
Numero doppio: € 9,00
Numero con allegato: € 8,50

Autorizzazione alla pubblicazione del Tribunale di Milano n. 647 del 17/11/2003.
Iscrizione al R.O.C. n. 11035 19/11/2003

© Copyright - Tutti i diritti di riproduzione o di traduzione degli articoli pubblicati sono riservati. Manoscritti, disegni e fotografie sono di proprietà di INWARE srl. È vietata la riproduzione anche parziale degli articoli salvo espressa autorizzazione scritta dell'editore. I contenuti pubblicitari sono riportati senza responsabilità, a puro titolo informativo.

Privacy - Nel caso la rivista sia pervenuta in abbonamento o in omaggio, si rende noto che i dati in nostro possesso sono impiegati nel pieno rispetto del D.Lgs. 196/2003. I dati trasmessi a mezzo cartoline o questionari presenti nella rivista, potranno venire utilizzati per indagini di mercato, proposte commerciali, o l'inoltro di altri prodotti editoriali a scopo di saggio. L'interessato potrà avvalersi dei diritti previsti dalla succitata legge. In conformità a quanto disposto dal Codice di deontologia relativo al Trattamento di dati personali art. 2, comma 2, si comunica che presso la nostra sede di Cormano Via Cadorna 27, esiste una banca dati di uso redazionale. Gli interessati potranno esercitare i diritti previsti dal D.Lgs. 196/2003 contattando il Responsabile del Trattamento Inware Srl (info@inwaredizioni.it).

RICHIESTE DI ASSISTENZA
Per richiedere assistenza o chiarimenti sugli articoli pubblicati, vi preghiamo di contattare direttamente l'autore. Se questo non fosse possibile, utilizzate il modulo di contatto che trovate sul nostro sito web www.farelettronica.com.

COLLABORARE CON FARE ELETTRONICA
Le richieste di collaborazione vanno indirizzate all'attenzione di Tiziano Galizia *(t.galizia@farelettronica.com)* e accompagnate, se possibile, da una breve descrizione delle vostre competenze tecniche e/o editoriali, oltre che da un elenco degli argomenti e/o progetti che desiderate proporre.

ELENCO INSERZIONISTI

Alterlogix ..... **pag. 27**
Via Giotto, 19 - 64026 Roseto degli Abruzzi (TE)
Tel 0858933615 - www.alterlogix.com

Artek Electronics Solution ..... **pag. 91**
Piazza Pirazzoli, 2 - 40020 Sasso Morelli (BO)
Tel 0542.643192 - www.artek.it

Blu Nautilus ..... **pag. 55**
Piazza Tre Martiri, 24 - 47900 Rimini
Tel 0541.53294 - www.blunautilus.it

Compendio Fiere ..... **pag. 87-107**
Via Mammianese, 53 - 51017 Pescia (PT)
Tel 0572.490868 - www.compendiofiere.it

Comune di Scandiano (ufficio fiera) ..... **pag. 67**
Piazza Prampolini, 1 - 42019 Scandiano (RE)
Tel 0522.764290 - www.fierascandiano.it

Consorzio Elettrimpex. ..... **pag. 31**
Via Console Flaminio,19 - 20134 Milano
Tel 02.210111230 - www.elettrimpex.it

Dinema ..... **pag. 15**
Via S. Polo, 183 - 25134 Brescia
Tel 030.2300492 - www.dinema.it

Eikon ..... **pag. 126**
Via Borgognina, 5 - 61030 Lucrezia (PS)
Tel 0721.877365 - www.eikonsite.it

E V R Electronics ..... **pag. 7-11**
Viale Kennedy, 96 - 20027 Rescaldina (MI)
Tel 0331.464893 - www.evr-electronics.com

Expofiere. ..... **pag. 95**
Casella Postale 50 - 44030 Ro Ferrarese (FE)
Tel 0532.832853 - www.expofiere.com

Futura Elettronica ..... **pag. 43-61-83-117-121**
Via Adige, 11 - 21013 Gallarate (VA)
Tel 0331.792287 - www.futuranet.it

Gedit ..... **pag. 73**
Via della Scuola, 128 - 06087 Ponte San Giovanni (PG)
Tel 075.5991028

Grifo ..... **II cop.**
Via dell'Artigiano, 8/6 - 40016 San Giorgio Di Piano (BO)
Tel 051.892052 - www.grifo.it

Idea Elettronica ..... **pag. 103**
Via S. Vittore 24/A - 21040 Oggiona con S.Stefano (VA)
Tel 0331.502868 - www.ideaelettronica.it

Millenium Dataware. ..... **pag. 39**
Corso Repubblica, 48 - 15057 Tortona (AL)
Tel 0131.860254 - www.mdsrl.it

Netwaves ..... **pag. 35**
Via Cadorna, 27/31 - 20032 Cormano (MI)
Tel 02.66504794 - www.netwaves.it

Precma ..... **pag. 33-115**
Via Fontanino 4 - 23871 Lomagna (LC) - IT
Tel 039.5300590 - www.precma.it

RS Components ..... **III cop - pag. 19**
Via M. V. De Vizzi, 93/95 - 20092 Cinisello Balsamo (MI)
Tel 02.660581 - www.rs-components.com

Scuola Radio Elettra ..... **IV cop.**
Via Biturgense, 104 - 00185 Cerbara di Città di Castello (PG)
Tel 075.862911 - www.scuolaradioelettra.it

S.V.M. Elettronica ..... **pag. 97**
Via Sempione, 24 - 21057 Olgiate Olona (VA)
Tel 0331.640569 - www.svmelettronica.com

Studio Fulcro ..... **pag. 65**
Piazzale Kennedy, 1 - 16129 Genova
Tel 010.561111 - www.studio-fulcro.it

Telecontrolli ..... **pag. 37**
Via Valla, 16 - 20141 Milano
Tel 02.84742360 - www.telecontrolli.com

Sommario

246
Dicembre
2005

Pratica

La tecnica DDS: Generatore di segnali a frequenza variabile — 22
di Salvatore Torrisi

Mikrobasic per PICmicro (ottava parte): Ambiente di sviluppo — 44
di Giovanni Di Maria

Biofeedback: Trasmissione a distanza — 58
di Vittorio Marradi

CPLD By Example (terza parte): Sistema hardware di sviluppo — 84
di Agostino Rolando

Usare PROTEUS (terza parte): ARES - Tecniche di base per il piazzamento e lo sbroglio di PCB — 108
di Maurizio Del Corso

pag. 44

Teoria

Elettronicando (undicesima parte): Circuiti reazionati, oscillatori e reti di sfasamento — 12
di Riccardo Nicoletti

L'amplificatore operazionale dalla A alla Z (nona parte): L'operazionale a reazione di corrente — 76
di Nico Grilloni

Alimentatori switching (dodicesima parte): Analisi di uno switching off-line (parte II) — 98
di Romano Bernarducci

Vitamina C (ventiquattresima parte): Sistemi operativi — 118
di Antonio Di Stefano

pag. 58

Risorse

Mailbox — 6
Prima pagina — 8
Notepad — 10
Gli appuntamenti di Dicembre 2005 — 56

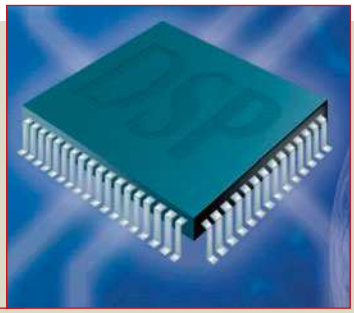
Riflettori su...
EIKON: "Wireless Made Easy" — 126

pag. 98

Speciale

Introduzione ai processori DSP
di Enrico Raffone

pag. 28



Scrivete a:
MAILBOX
REDAZIONE DI
FARE ELETTRONICA
Inware s.r.l.
Via Cadorna, 27/31
20032, Cormano (MI)

Oppure inviate un'email a:
mailbox@farelettronica.com

Mailbox

Questa rubrica ospita le richieste più interessanti pervenute dai lettori. Per quanto possibile verrà data risposta a tutte le richieste pervenute via email.

MICROFONO WIRELESS

Mi piacerebbe veder pubblicato lo schema di un trasmettitore microfonico FM nella banda degli 88MHz. Vorrei integrare questo trasmettitore nel mio robot da minisumo per cui dovrebbe essere il più piccolo possibile.

Gerardo Seglie

Il più semplice trasmettitore microfonico FM realizzabile nella banda degli 88MHz è quello riportato in figura 1. Il segnale del microfono viene inviato alla base di T1 che, con R1, R2, C1 e C6 costituisce uno stadio di amplificazione ad emettitore comune. Il segnale così amplificato, raggiunge la base di T2 variandone il punto di lavoro, quindi modulando la frequenza di oscillazione. L1 e C4 costituiscono il circuito accordatore mediante il quale si può regolare la frequenza della portante (a tale scopo è possibile impiegare una induttanza montata su nucleo mobile in ferrite, o un compensatore al posto di C4). Come antenna andrà benissimo uno spezzone di cavo della lunghezza di 21cm. L'alimentazione a 9V può essere ricavata impiegando una semplice batteria alcalina.

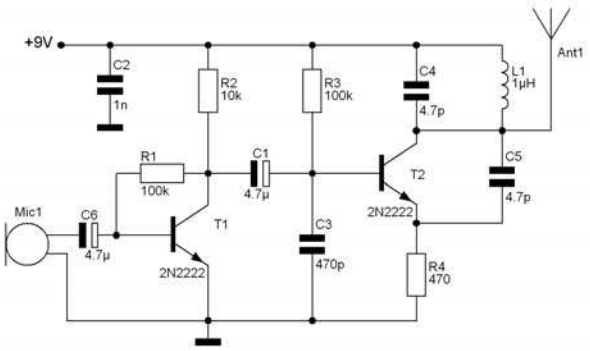


Figura 1 *Microfono wireless*

DUPLICATORE DI FREQUENZA

Per creare un effetto luci avrei bisogno di un circuito per raddoppiare la frequenza di un segnale ad onda quadra. Ho visto in rete molti circuiti divisori, ma mi è difficile recuperare lo schema di un duplicatore, potete aiutarmi?

Mario Vandelli

Un duplicatore di frequenza può essere ottenuto con il circuito di figura 2 che impiega un comunissimo 4069 contenente sei inverter CMOS. Gli impulsi da duplicare, la cui ampiezza deve essere compresa tra 3 e 15V, vengono applicati all'ingresso della porta P1 la quale li inverte e li presenta a P2 per una successiva inversione in modo da ottenere due segnali quadrati sfasati di 180°. Tali segnali vengono derivati dalle reti RC, invertiti e squadrati dalle porte P3 e P4 e sommati dalla rete D1, D2, R3 e P5. Il range di frequenze a cui può operare il circuito è molto vasto, basterà tener conto dell'impulso minimo sopportabile dalle porte P3 e P4. Con lo schema di figura 2 si possono usare in ingresso segnali fino a 500KHz.

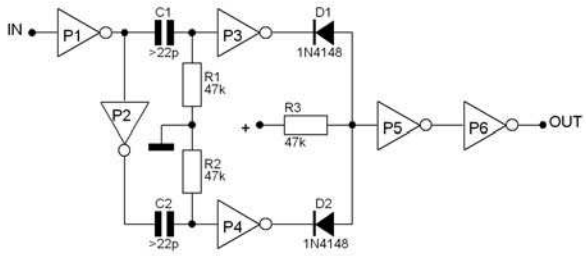


Figura 2 *Il duplicatore di frequenza*

“Richieste, chiarimenti, dubbi e commenti dai lettori”

UN INTERRUTTORE PARTICOLARE...

Avrei la necessità di accendere due led in funzione della posizione di un deviatore e contemporaneamente chiudere un contatto per un centinaio di millisecondi indipendentemente dalla posizione del deviatore stesso. In questo modo potrei simulare la pressione di un tasto su una tastiera semplicemente commutando manualmente il deviatore.

Michele Piu

Il circuito richiesto è riportato nella figura 3. Esso è costituito da un multivibratore monostabile a porte CMOS la cui durata dell'impulso in uscita è funzione del prodotto tra R e C secondo la formula T=0,693RC. Normalmente l'uscita dell'ultima porta è a livello basso per cui il transistor è interdetto ed il relè non è eccitato. Durante l'impulso sull'uscita il transistor va in saturazione portando il relè in eccitazione per tutta la durata dell'impulso. La posizione del deviatore determina l'accensione del relativo LED. La resistenza da 1K ha un duplice scopo: quello di limitare la corrente nel LED a poco più di 10mA e quello di mantenere a livello basso l'ingresso 2 della prima porta nel caso in cui nessun LED sia connesso. Nella condizione di riposo infatti il multivibratore richiede un livello alto sull'ingresso (garantito dal fatto che il LED acceso comporta una caduta di tensione di circa 1V) mentre per l'innesco dell'impulso di uscita è necessario un impulso negativo di breve durata sull'ingresso. Spostando il deviatore da una posizione all'altra ci sarà un breve intervallo di tempo in cui entrambi i contatti sono aperti e in questo frangente la resistenza porta a zero l'ingresso 2 provocando l'innesco dell'impulso. Il diodo in parallelo ad R garantisce un tempo di ripristino molto veloce in modo da poter richiedere un nuovo impulso subito dopo il termine del precedente.

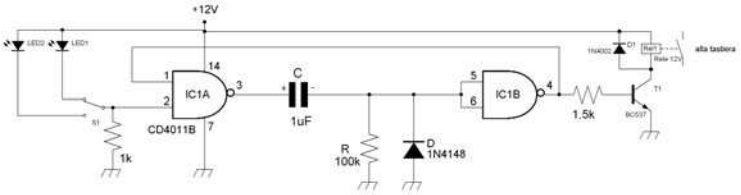
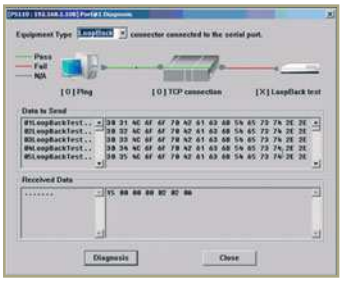


Figura 3 *Un interruttore particolare*

Rubrica di notizie e novità dal mondo dell'elettronica.

Prima pagina

246-01 SENA ANNUNCIA LA VERSIONE 2.0 DI HELLODEVICE MANAGER



Sena annuncia la versione 2.0 di HelloDevice Manager, l'utility utilizzata per il controllo di diversi dispositivi, questa nuova versione è corredata di nuove funzioni user-friendly che semplificano la gestione della famosa linea di device server HelloDevice. Le principali funzioni di HelloDevice Manager sono:

- Provare i Sena Device Server connettendosi all'interfaccia di gestione direttamente, usando il web, telnet o SSH.
- Aggiungere alcuni promemoria direttamente nell'applicazione.
- Sono disponibili diverse funzioni diagnostiche molto semplici da usare che consentono di provare il trasferimento dei dati tra il device server, l'host remoto, i device server seriali e i dispositivi seriali connessi ad ogni sezione, usando un'interfaccia grafica. Inoltre è possibile controllare se la connessione alla rete è in uno stato normale, oltre a poter inviare o ricevere dati tra il PC host e i dispositivi seriali.
- È possibile controllare lo stato dei pin in real-time.
- È possibile controllare i log SNMP in un'apposita finestra.

246-02 GLOBAL RANGE: RITORNO AL FUTURO PER RS

Ermanno Maffé, amministratore delegato RS Italia, ha recentemente annunciato la partenza del Global Range, accessibile dal sito rswww.it.

Si tratta di 80.000 nuovi prodotti disponibili in 4 gg. direttamente tramite la consociata americana Allied, con sede a Dallas e 20.000 nuovi prodotti dai fornitori strategici Tyco e Molex.

Grazie al Global Range, il catalogo RS raggiunge l'impressionante cifra di 220.000 prodotti, di cui oltre il 30 percento gestiti attraverso il sito web. RS ha iniziato molti anni fa in Italia col concetto dei piccoli volumi e la grande quantità di clienti. Sono infatti oltre 50 mila i clienti che acquistano abitualmente da RS, per le esigenze di pre-produzione ancora decisamente in carico al mercato europeo, mentre le grandi produzioni stanno rapidamente prendendo la strada dell'Est. Il mondo educational complessivamente rappresenta per RS il primo cliente, grazie al Global Range e la consulenza sui prodotti, uno dei benefici tipici di RS, gli obiettivi di crescita per il prossimo anno sono inaspettatamente a due cifre.

Essendo un distributore, RS non progetta né produce i prodotti. L'azienda collabora strettamente con i propri fornitori per assicurare la piena conformità degli articoli presenti a catalogo alle Direttive RoHS e RAEE e, per questo, ha istituito dei "team multifunzionali" incaricati di gestire e coordinare tutte le attività correlate.

Già da oggi, RS offre alla propria clientela un servizio di impareggiabile qualità, che consiste nel fornire un supporto professionale a tutti coloro che si cimentano nella comprensione e nell'implementazione delle due normative. Inoltre, su RS Online sono già stati introdotti dei simboli che indicano, per ciascun prodotto, lo stato di conformità o non conformità.

PER SAPERNE DI PIÙ

Per approfondire le notizie riportate in questa rubrica, visitate il sito **www.farelettronica.com/primapagina** e seguite le istruzioni.
In alternativa potete scrivere a:

Inware Edizioni
Servizio Prima Pagina
Via Cadorna 27 - 20032 Cormano (MI)

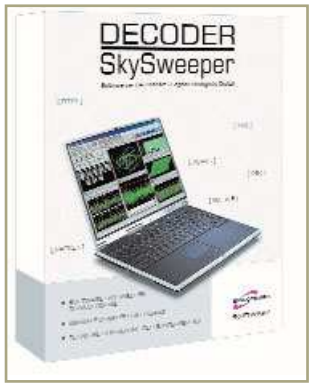
Indicando il codice riportato accanto al titolo della notizia (esempio 246-03).

246-03 KEVIN SCHURTER DISTRIBUISCE IDT

IDT Integrated Device Technology e Kevin Schurter S.p.A. annunciano di aver firmato un contratto di collaborazione che vede Kevin Schurter S.p.A. come nuovo Distributore di IDT per il mercato italiano. Questa scelta comune si integra perfettamente con le strategie di entrambe le Società; infatti IDT, Società leader nella produzione di Memorie Statiche, FIFO, Multi-port, Timing Solution, Network Search Engine e Serial Switching Products, ha da sempre optato in ogni Paese per la soluzione di affiancare un distributore broad line a uno che fa promozione tecnica e dell'attività di design uno dei suoi maggiori punti di forza. Questa politica di IDT si sposa perfettamente con la strategia che da sempre Kevin Schurter ha adottato per approcciare il mercato italiano. Kevin Schurter, Società facente parte al 100% del Gruppo svizzero Schurter, leader nella produzione di Fuse, fuse holder, power entry, key and keyboard, oltre ad essere responsabile per i prodotti costruiti dalla casa madre per il mercato italiano e per i mercati dei Paesi del Sud-Est, è da anni presente in tutta Italia quale Società specializzata nella distribuzione di prodotti di nicchia ed ad alta tecnologia sia nell'ambito degli attivi, che in quello dei passivi e della visualizzazione. Di fatto la strategia di Kevin Schurter è basata su uno staff commerciale composto da venditori, sia esterni che interni, con una solida preparazione tecnica che permette loro già dalle prime fasi di trattativa di dialogare con i clienti non solo a livello commerciale. A questa struttura prettamente di vendita si affianca uno staff di Field Application Engineers totalmente dedicato sia alla promozione tecnica dei prodotti, sia all'affiancamento e supporto, là dove necessario, degli uffici di progettazione dei clienti supportando i progettisti stessi nello sviluppo dei loro prodotti. Con queste basi comuni di ideologia e strategia, la Partnership fra IDT e Kevin Schurter rappresenta la soluzione ottimale per entrambe le Società finalizzata a conseguire il successo sul mercato italiano che i prodotti IDT meritano.

246-04 DECODER SKYSWEEPER, SOFTWARE SU CD-ROM

SkySweeper è un programma che implementa le più avanzate funzioni DSP di elaborazione digitale dei segnali al fine di ricevere e trasmettere nella maggior parte dei modi e codici operanti su bande di frequenza HF e V/UHF usando quale dispositivo di ingresso/uscita la sola scheda audio del computer.



SkySweeper è divenuto a livello mondiale una delle piattaforme software preferite dagli utenti privati come governativi per l'ottima capacità di rilevare segnali con basso rapporto segnale/rumore e per la flessibilità di configurazione che ne rende l'impiego particolarmente efficace anche nell'elaborazione di codifiche inusuali oppure non standard. Il software supporta le emissioni nei modi: 2MFSK16, 4MFSK16, ACARS, AMTOR, CW, DGPS, DSC, FELD-HELL, GMDSS, HFDL, HF-FAX, MARTIN 1, MARTIN 2, MFSK16, MIL-ALE, NAVTEX, PACKET AX25, PACTOR, PSK105-HELL, PSK125, PSK245-HELL, PSK31, PSK63, QPSK, RTTY, SCOTTIE 1, SCOTTIE 2, SELCAL, SITOR-A, SITOR-B, SkyBoost, SSTV, SYNOP, WEFAX.

A complemento del manuale, dove trovano descrizione le configurazioni per tutti i modi citati, delle note tecniche esaminano le diverse modalità per individuare e ridurre le interferenze RFI generate dal computer oppure da altri apparati che vanno a degradare la qualità della ricezione.

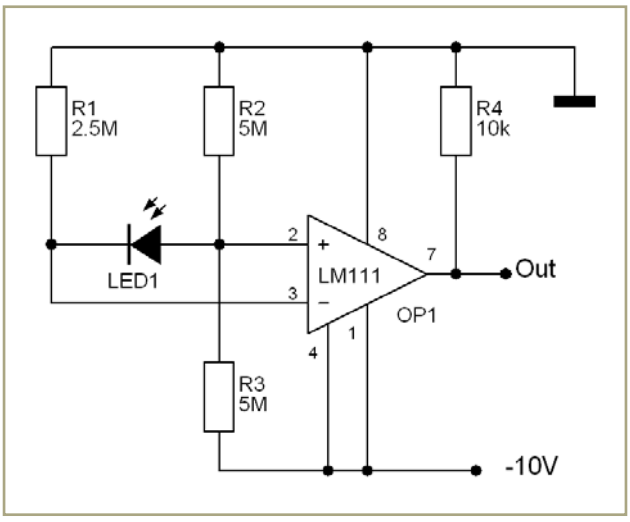
Dal blocco note di Fare Elettronica una raccolta di idee da tenere sempre a portata di mano.

Notepad

Questa rubrica ha lo scopo di fornire degli schemi applicativi o idee di progetto dei componenti elettronici più interessanti, selezionati per voi dalla redazione. Tutti gli schemi presentati sono elaborazioni di quelli ufficiali proposti dai produttori nella documentazione ufficiale.

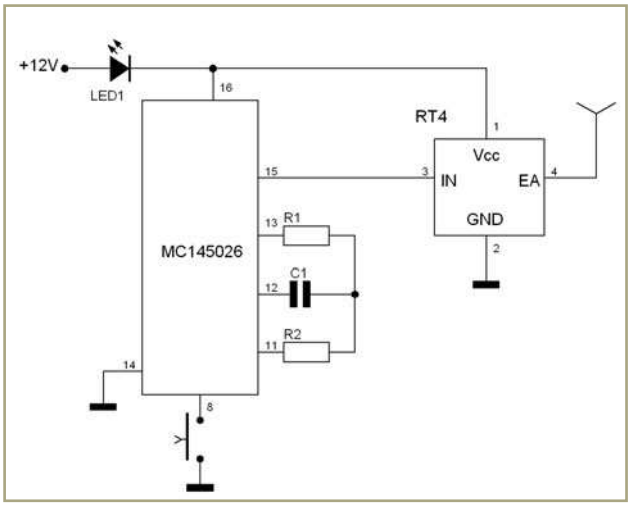
LEVEL DETECTOR PER FOTODIODO

Una semplice soluzione per determinare lo stato di un fotodiodo alimentato con una tensione negativa di -10V.
Il circuito impiega un LM111 montato come comparatore la cui tensione di riferimento è fissata a -5V dal partitore resistivo costituito da R2 ed R3.
L'uscita del circuito cambia stato quando la corrente nel fotodiodo raggiunge 1µA.



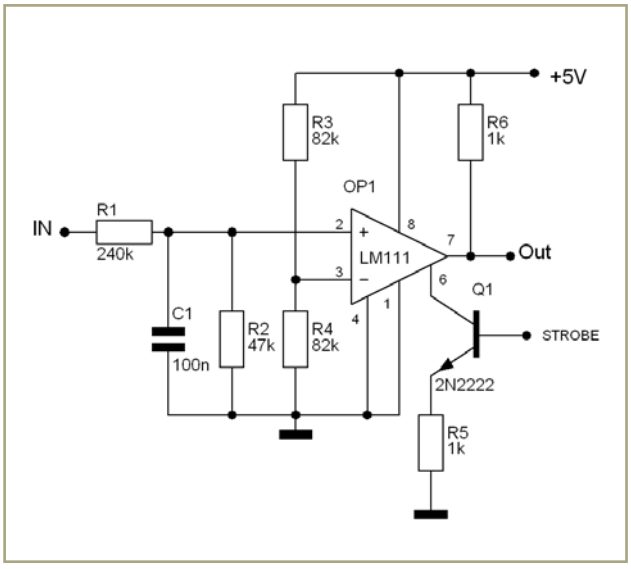
TRASMISSIONE IN RADIOFREQUENZA SU 433MHZ

Sono necessari solamente pochi componenti per trasmettere codici i radiofrequenza sulla banda 433,92MHz. In figura lo schema applicativo di un modulo ibrido con risonatore SAW ed antenna esterna. I valori dei componenti dipendono dalla frequenza di lavoro.



INTERFACCIA HIGH VOLTAGE-TTL

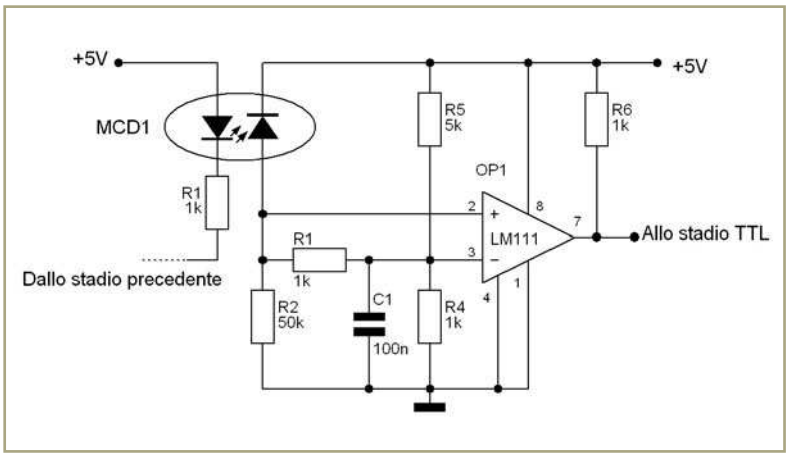
Il circuito in figura mostra una applicazione



dell'amplificatore operazionale LM111 come interfaccia tra circuiti operanti ad elevati valori di tensione e circuiti digitali basati su logica TTL.
Il segnale di ingresso può essere compreso tra 0V e 30V ed il partitore costituito da R1 ed R2 provvede a restringere il range tra 0 e 5V.
R3 ed R4 fissano la tensione di riferimento a 2,5V in modo che il comparatore commuti quando l'ingresso supera i 15V.
La coppia Q1 ed R5 provvede a fornire un ingresso di STROBE che, se portato a 5V, pone a livello alto l'uscita del circuito indipendentemente dal valore assunto dal segnale di ingresso.

ISOLAMENTO GALVANICO TRA STADI TTL

Ecco un esempio di come utilizzare un LM111 come stadio di isolamento ottico tra due stadi TTL.
L'isolamento avviene grazie alla presenza del fotoaccoppiatore a fotodiodo MCD1 il cui LED viene gestito dallo stadio pilotante.
Il circuito può funzionare correttamente fino a 1MHz, mentre per frequenze inferiori ai 200KHz la coppia R3, C1 può essere omessa. Il fotoaccoppiatore può anche essere costituito banalmente da un LED ed un fotodiodo accoppiati otticamente tra loro: in questo caso sarà necessario assicurarsi che il fotodiodo sia investito dalla sola luce del LED.



Il guadagno ed altre caratteristiche di un amplificatore dipendono fortemente dai parametri dei dispositivi attivi che lo compongono, e quindi dal loro punto di riposo. Il punto di riposo è tuttavia suscettibile delle variazioni di temperatura e delle dispersioni delle caratteristiche dei singoli dispositivi; si cerca quindi di rendere il guadagno dell'amplificatore indipendente dai parametri elettrici del dispositivo attivo, facendo in modo che esso dipenda esclusivamente da componenti molto stabili, come i resistori. Si realizza tutto questo ricorrendo alla "retroazione", che porta numerosi vantaggi. Dal principio della retroazione si illustra come ottenere delle oscillazioni spontanee, introducendo il criterio di Barkhausen.

LA REAZIONE

Un concetto particolarmente importante in elettronica analogica è quello di "*reazione*" o, in inglese, "*feedback*". Il concetto deriva dalla Teoria dei Controlli Automatici ed è largamente adottato sia in meccanica che in elettronica.
Un sistema *reazionato* consente di operare un certo controllo sull'uscita e regolare di conseguenza l'ingresso, in modo da conferire al tutto un opportuno grado di stabilità.
Si pensi ad esempio ad un tostapane, ad uno scaldabagno, ad uno sciacquone, tanto per fare esempi tratti dal quotidiano; essi sono tutti sistemi dove l'uscita è monitorata per regolare in modo opportuno l'ingresso.
Gli amplificatori elettronici utilizzano la reazione per mantenere costante nel tempo il rapporto tra il segnale di uscita e quello di ingresso, in modo da riprodurre questo nel modo più fedele possibile.
Banalmente, potremmo dire che una parte del segnale di uscita di un dispositivo viene prelevata, elaborata e confrontata con il segnale di ingresso.
Questa tecnica, che viene poi suddivisa in "*reazione negativa*" e "*reazione positiva*", serve per conferire al sistema che la utilizza vantaggiose caratteristiche per quanto riguarda, ad esempio, la stabilità in frequenza o per rendere alcuni parametri elettrici particolarmente appetibili.
Trascurando le ipotesi che stanno alla base dello studio della reazione ed i dettagli della trattazione (che necessiterebbe di ben altro spazio), vediamo di chiarire meglio quanto abbiamo detto in questa introduzione.

UN SISTEMA RETROAZIONATO

Riferiamoci ad una categoria di sistemi dove la reazione è largamente usata: gli amplificatori.

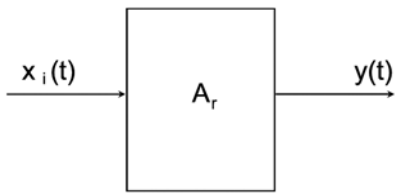


Figura 1 *Un amplificatore retroazionato visto dall'esterno*

Circuiti reazionati, oscillatori e reti di sfasamento

di Riccardo Nicoletti
(r.nicoletti@farelettronica.com)

Supponiamo che l'amplificatore di figura 1 contenga un sistema retroazionato al suo interno.
Indichiamo con $x_i(t)$ il *segnale di ingresso*, espresso in funzione del tempo; y(t) è *l'uscita* espressa anch'essa in funzione del tempo. Le lettere minuscole indicano in generale grandezze variabili scritte come funzioni del tempo.
Per ricordarci che esso è retroazionato, indichiamo il suo guadagno con A_r utilizzando il pedice "r".
L'interno del sistema "*retroazionato*" può essere scomposto e rappresentato come in figura 2. Il segnale di ingresso x_i entra nel blocco amplificatore e genera il segnale di uscita y; questo segnale viene prelevato da una "*rete di reazione*" che lo trasforma in x_f (il pedice "f" sta per "feedback"); questo segnale x_f finisce in un blocco di confronto dove va a sottrarsi con x_i generando il "segnale errore" $e=x_i-x_f$ che va in ingresso all'amplificatore.
A cosa può servire tutto questo?
Facciamo un esempio. Supponiamo che, per una causa fortuita, l'uscita dell'amplificatore aumenti bruscamente; il segnale y è dunque cresciuto e conseguentemente aumenta anche x_f; se cresce x_f diminuisce e(t), che costituisce l'ingresso dell'amplificatore e quindi y decresce a sua volta. Questo è un semplice effetto della reazione negativa: una maggiore stabilità.
I sistemi retroazionati hanno generalmente delle prestazioni non eccezionali, ma possiedono maggiore stabilità rispetto ai sistemi privi di reazione. Questo vale anche per sistemi meccanici.
Dal rapporto y(t)/x(t) si ricava la funzione

$$A_r = \frac{A}{1+\beta A}$$

dove A_r è il *guadagno a catena chiusa* (closed loop gain), A è il guadagno dell'amplificatore interno (ovvero il guadagno a catena aperta, "open loop gain") e β indica la funzione di trasferimento del blocco di reazione (figura 3).
Il termine 1+βA viene chiamato *fattore di retroazione*. Nel caso in cui il blocco di confronto operi la differenza tra $x_i(t)$ ed $x_f(t)$ si parla di "*reazione negativa*", mentre si parla di "*reazione positiva*" se $x_i(t)$ e $x_f(t)$ vengono sommati nel blocco di confronto. Nel caso di reazione positiva la relazione tra ingresso e uscita diventa:

$$A_r = \frac{A}{1-\beta A}$$

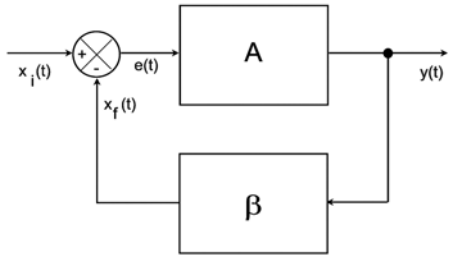


Figura 3 *Schema a blocchi di un sistema con reazione negativa*

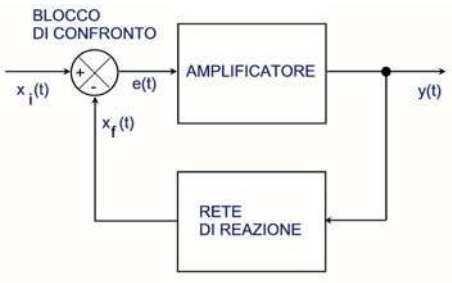


Figura 2 *Schema a blocchi di un sistema con reazione negativa*

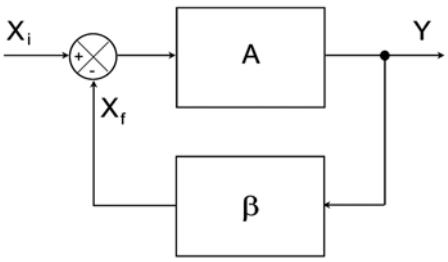


Figura 4 *Schema a blocchi di un sistema con reazione, rappresentato con le trasformate dei vari segnali*

LA FUNZIONE DI TRASFERIMENTO

Una precisazione importante. Per ricavare la funzione A_r si è fatto il rapporto tra y(t) ed x(t), commettendo un errore che gli accademici di Controlli Automatici potrebbero non perdonarci. Lo abbiamo fatto a scopi didattici. La funzione A_r è la *funzione di trasferimento* dell'amplificatore retroazionato, pertanto le grandezze che la compongono (ed essa stessa!) devono essere valutate nel dominio delle trasformate di Laplace o di Fourier. Tali trasformate, indicate con lettera maiuscola, sono funzioni di variabile complessa, che necessitano di un'ampia trattazione matematica. La trasformata di Laplace è espressa in funzione della variabile complessa "s", mentre quella di Fourier è funzione di "jω". Entrambe esprimono il comportamento di un sistema in funzione della frequenza anziché del tempo. A rigore quindi nello schema a blocchi dovremmo riportare le relative trasformate di tutti i segnali (figura 4).

I VANTAGGI DELLA REAZIONE NEGATIVA

Riprendendo il discorso fatto nella nostra introduzione, i vantaggi della reazione negativa arrivano quando si riesce a rendere $\beta A \gg 1$. In tal caso infatti si ha:

$$A_r = \frac{A}{1+\beta A} \approx \frac{A}{\beta A} = \frac{1}{\beta}$$

e quindi il guadagno a catena chiusa dipende solo dal blocco β. Teniamo presente che il blocco di reazione è costituito solo da *componenti passivi* e pertanto, adottando componenti di precisione, si riesce ad ottenere A_r estremamente stabile.

Con queste premesse si ottengono i numerosi vantaggi della reazione negativa: una stabilizzazione del punto di riposo del dispositivo attivo, una stabilizzazione del guadagno, una modifica della resistenza di ingresso e di uscita dell'amplificatore, una modifica della risposta in frequenza, la riduzione degli effetti di un disturbo e la riduzione della distorsione.

Ovviamente tutto questo si paga. Come? È presto detto: se $\beta A \gg 1$ risulta senza dubbio $A_r \ll A$ e quindi tutto si ottiene a spese del guadagno.

LA REAZIONE POSITIVA E GLI OSCILLATORI

Abbiamo visto come la reazione negativa sia in un certo modo "salutare" per un amplificatore. La reazione positiva porta invece ad instabilità e non viene usata per migliorare le caratteristiche di fedeltà di un amplificatore, poiché essa può spesso portare alla nascita di *fenomeni oscillatori* del tutto inaccettabili. Viene invece utilizzata quando le oscillazioni sono volute, quindi per la realizzazione di oscillatori sinusoidali.

Vediamo come si utilizza la reazione positiva a questo scopo.

LA CONDIZIONE DI BARKHAUSEN

A cosa serve la funzione di trasferimento A_r?

Essa è definita come il rapporto tra il segnale di uscita Y (o meglio, la trasformata di Laplace del segnale di uscita) ed il segnale di ingresso X_i (trasformato nel dominio di Laplace).

$$A_r = \frac{Y}{X_i}$$

Noto il segnale di ingresso X_i, essa serve dunque a ricavare il segnale di uscita Y:

$$Y = A_r X_i$$

e da questa espressione nasce una considerazione: se rendo il segnale di ingresso X_i quasi zero, è possibile ottenere un segnale di uscita non nullo?

$$Y = \frac{A}{1-\beta A} \times \text{quasi zero}$$

se il denominatore della frazione diventa infinitamente piccolo, la matematica ci insegna che la frazione assume valori tendenti ad infinito, ovvero molto grandi; sempre dalla matematica sappiamo che il prodotto tra un numero che tende ad infinito ed un numero infinitamente piccolo dà un risultato diverso da zero.

Questo vuole dire che Y non è zero, quindi ottengo *qualcosa* in uscita. Ma cosa significa avere X_i quasi zero? Se non poniamo segnale di ingresso all'amplificatore (figura 5) possiamo assimilare il segnale di ingresso al solo "*rumore*" che inevitabilmente il sistema sente in ingresso; i transitori di accensione del dispositivo ed il rumore termico dei componenti sono sufficienti ad innescare l'oscillazione.

La Passione
per il servizio completo

- **P**rogettazione hardware e software
- **C**odesign & coengineering
- **F**ornitura in conto pieno. Gestione autonoma dei materiali
- **P**roduzione SMT THT COB micro BGA
- **A**ssemblaggio in cabinet
- **C**ollaudi ICT, FUNZIONALI, DEDICATI, TEST OTTICI
- **C**amere di RUN-IN, CELLE CLIMATICHE
- **G**estione del Post vendita

IL VOSTRO PARTNER PER AVERE SOLUZIONI A 360° NELL' ELETTRONICA E NELL' AUTOMAZIONE INDUSTRIALE

ELETTRONICA ED AUTOMAZIONE INDUSTRIALE

Via S.Polo, 183 - 25134 Brescia (Italy)
Tel. +39 030 2300492 - Fax. +39 030 2300833
Internet: www.dinema.it - mail: dinema@dinema.it

dinema
ELECTRONICS

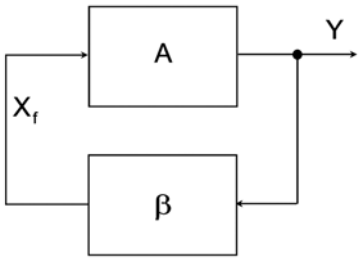


Figura 5 *Sistema con reazione con ingresso nullo*

Imporre la condizione 1-βA=0 significa imporre βA=1. Tale condizione prende il nome di "*condizione di Barkhausen*", dal nome del fisico tedesco (figura 5a) Heinrich Georg Barkhausen (1881-1956) che studiò, tra le altre cose, la generazione di oscillazioni elettriche spontanee.

Poiché i termini A e β sono funzioni della frequenza, e contengono quindi la variabile frequenza, è necessario che la condizione di Barkhausen sia verificata ad una sola frequenza, che sarà quella alla quale l'oscillatore funzionerà. Parlando di una sola frequenza ci è chiaro che l'oscillazione sarà di tipo sinusoidale.

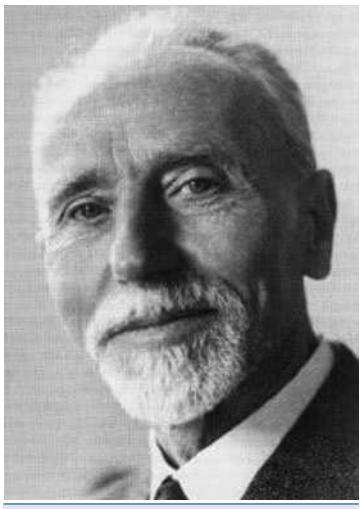


Figura 5a *Il fisico tedesco Heinrich Georg Barkhausen*

LA NATURA COMPLESSA DELLA FUNZIONE DI TRASFERIMENTO

Facciamo un passo indietro. Abbiamo nominato le "*funzioni di trasferimento*", che sono legami ingresso-uscita, scritti nel campo dei numeri complessi, sia nel dominio di Laplace che di Fourier.
Questo vuol dire che il prodotto βA possiede di per sé una *parte reale* ed una *parte immaginaria*, quindi un *modulo* ed un *argomento*.
Scrivere βA=1, significa imporre che il prodotto βA sia un numero reale pari ad 1; ulteriormente questo significa imporre che il prodotto βA (che è una grandezza complessa) possieda modulo pari ad 1 ed argomento nullo, ovvero:

$$\beta A = 1 \rightarrow |\beta A| = 1 \text{ e } \angle\beta A = 0°$$

Il prodotto βA si chiama "*guadagno ad anello aperto*" della catena β-A, ed è il guadagno che otteniamo se immaginiamo di aprire l'anello che contiene i due blocchi (Figure 6-7).

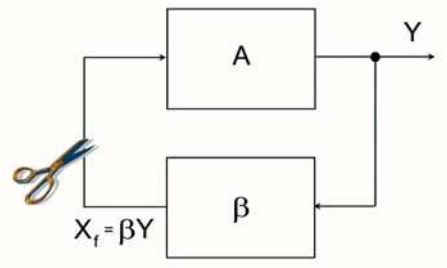


Figura 6 *Apriamo l'anello in assenza di segnale di ingresso*

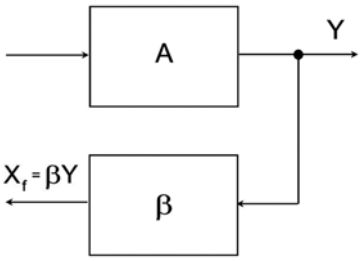


Figura 7 *Catena aperta in assenza di segnale di ingresso*

Se il modulo è 1 vuol dire che se il blocco A amplifica di 100, il blocco β deve attenuare di 100 in modo che il loro prodotto sia unitario; analogamente, se il blocco A sfasa di 30°in anticipo, il blocco β deve sfasare di 30°in ritardo, e così via. Spero di non avervi confuso le idee.

E c'è di più.
Generalmente il blocco β è una rete elettrica passiva, e quindi ad essa sono delegati valori di amplificazione sempre minori dell'unità.
Per realizzare un oscillatore sinusoidale è dunque necessario costruire un sistema con reazione positiva che, alla sola frequenza di oscillazione desiderata, soddisfi due condizioni:

1. Lo sfasamento complessivo lungo l'intera maglia deve essere nullo, cioè $\angle\beta A = 0°$(questa condizione determina la frequenza di oscillazione).
2. Il guadagno complessivo di tensione lungo

l'intera maglia deve essere unitario, cioè $|\beta A| = 1$ (questa condizione serve per dimensionare i vari componenti).

Affinché la frequenza alla quale sono soddisfatte le due condizioni sia unica è necessario che la rete di reazione sia selettiva e contenga quindi elementi reattivi.
Negli oscillatori che vedremo il blocco A sarà un amplificatore a larga banda ed il blocco β conterrà una rete passiva con elementi reattivi.
L'oscillatore deve "*autoinnescare*" l'oscillazione e poi "*autosostenerla*". Per far sì che l'oscillazione si autoinneschi è necessario imporre $\angle\beta A=0°$ ed $A>1$ all'inizio in modo che il rumore presente in ingresso venga amplificato moltissimo, e poi il mantenimento della condizione di Barkhausen provvederà alla stabilizzazione dell'oscillazione.

Da quanto detto gli oscillatori riescono a creare un'oscillazione in uscita pur in assenza di segnale di ingresso. Da dove si ricava l'energia associata al segnale di uscita?
La reazione è forse capace di creare qualcosa dal nulla?

OSCILLATORE A SFASAMENTO

Nell'oscillatore a sfasamento il blocco A è costituito da un amplificatore invertente realizzato con un operazionale (figura 8); questo amplificatore sfasa di 180°. Per riportare il segnale in fase tramite una rete passiva un solo circuito RC (o RL) non è sufficiente poiché arriva a sfasare meno di 90°, non ne sono sufficienti nemmeno due, ma ne occorrono tre, per riportare a 0°la fase lungo l'intera maglia (figura 9).
La condizione di Barkhausen è verificata per $A_v=29$. Nel caso di celle RC si ha:

$$f_o = \frac{1}{2\sqrt{6}\pi RC}$$

Una simulazione dell'oscillatore a sfasamento di figura 8 è riportata in figura 10.
Esistono alcune varianti allo schema di principio riportato in figura 8, come ad esempio quella di figura 11, dove sono state eliminate due resistenze.
Per il circuito di figura 11 la condizione di innesco è soddisfatta quando $R_f=12R$, e si ottiene

$$f_0 = \frac{1}{2\sqrt{3}\pi RC}$$

e con i componenti di figura 11 si ricava f_0 circa pari a 2298 Hz (figura 12).
La formula che fornisce f_0 è valutata ipotizzando che le celle RC non si carichino a vicenda; in

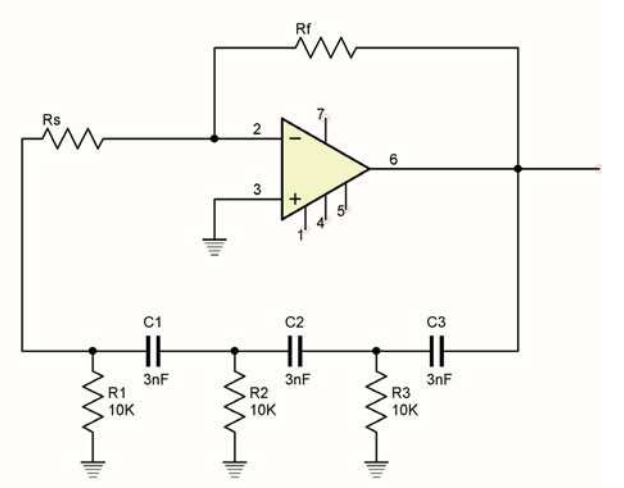


Figura 8 Schema di principio dell'oscillatore a sfasamento

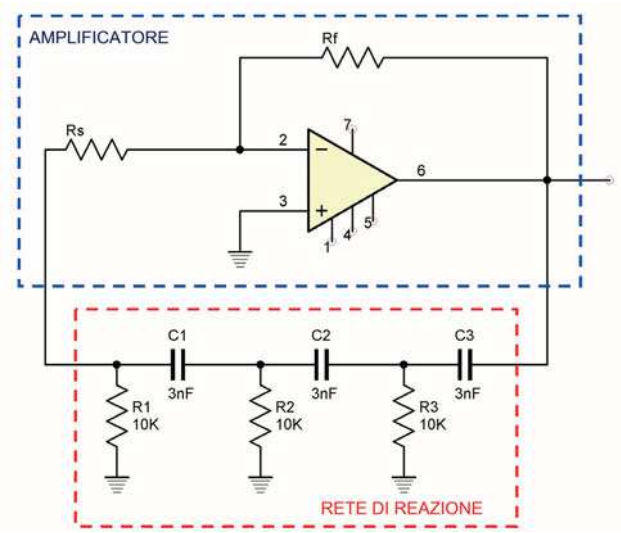


Figura 9 I blocchi che compongono l'oscillatore a sfasamento

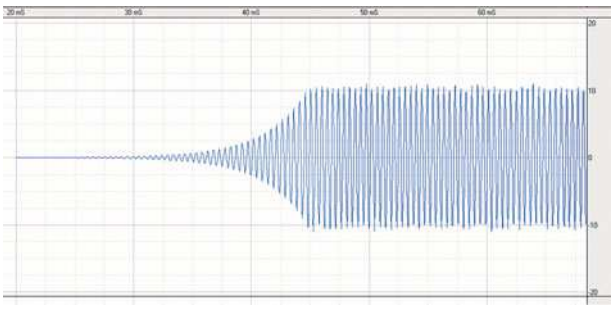


Figura 10 Simulazione per l'oscillatore a sfasamento di figura 9, con Rs=10kΩ ed Rf=55kΩ.

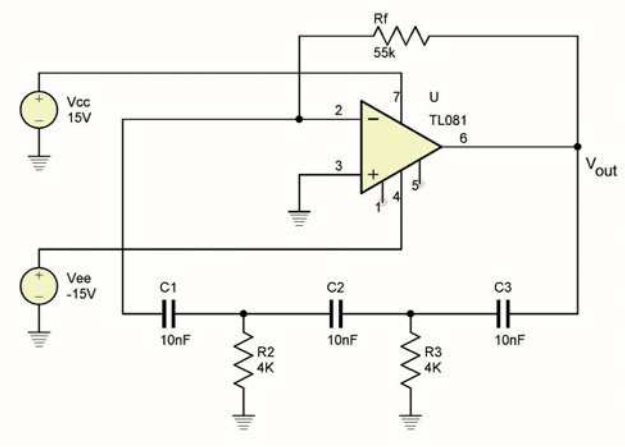


Figura 11 *Variante per l'oscillatore a sfasamento*

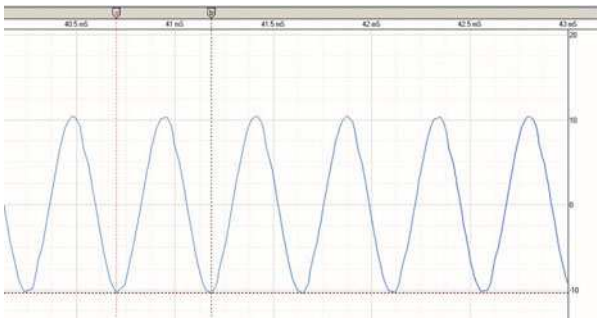


Figura 12 *Simulazione dell'oscillatore a sfasamento di figura 11*

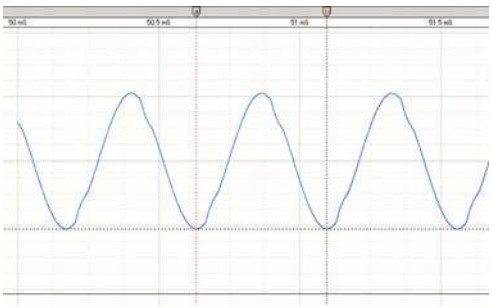


Figura 13 *Verifica della frequenza di oscillazione dell'oscillatore a sfasamento di figura 11*

realtà questo non è vero, quindi l'espressione di f_0 si ritiene approssimata. Si può ricavare il valore reale della frequenza di oscillazione sfruttando i *cursori* del simulatore (figura 13); leggendo il valore del periodo T dell'onda si ricava facilmente la frequenza $f_0=1/T$.

Si noti che nel circuito di figura 11 si è utilizzato un TL081, cioè un "Wide Bandwidth JFET Input OA" con slew-rate pari a 13 V/µs.

Per la figura 13 si legge dal simulatore T=461.6 µs, a cui corrisponde f_0=2166 Hz.

Provate a sostituire il TL081 con il uA741 per vedere se le prestazioni dell'oscillatore ne risentono.

Progettare un oscillatore sfruttando la condizione di Barkhausen è un'impresa alquanto delicata; ci sono numerosi accorgimenti da seguire per ottenere un circuito con prestazioni soddisfacenti. Citiamone alcuni:

1. Per frequenze dell'ordine delle decine di kilohertz occorre scegliere un A.O. con uno slew-rate abbastanza elevato; il solito uA741 non è molto adatto.
2. La condizione di Barkhausen con la rete di reazione composta da celle RC viene studiata supponendo che le tre celle non si carichino a vicenda, e questo in realtà, come già detto, non è vero; è possibile rendere più reale questa ipotesi separando le celle con degli inseguitori (figura 14).
3. L'uscita dell'oscillatore satura verso la tensione di alimentazione; esistono degli accorgimenti, come ad esempio l'aggiunta di due diodi Zener, per ottenere un'uscita con valore di tensione inferiore.
4. Esistono inoltre dei sistemi per ottenere un controllo automatico del guadagno.

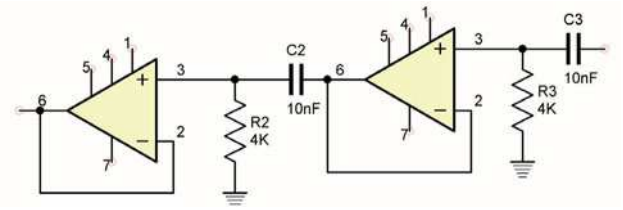


Figura 14 *Gli inseguitori separano tra loro le celle CR*

Oscillatore a ponte di Wien

Questo oscillatore si basa sul circuito a "*ponte di Wien*", elaborato da Max Wien nel 1891. Il circuito a ponte è uno dei metodi utilizzati per misure di capacità.

Max Wien (1866 - 1938) era un fisico tedesco, direttore dell'istituto di fisica dell'università di Jena.

Negli anni compresi tra il 1906 e 1909 inventò un generatore di onde elettromagnetiche leggermente attenuate che chiamò "*Löschfunkensender*" e che fu usato anche sul Titanic. Nel 1891 ideò l'oscillatore a ponte (che porta tuttora il suo nome), ma che non poté essere praticamente realizzato fino al 1939.

All'epoca non erano ancora stati inventati i dispositivi attivi, e quindi l'oscillatore non poté essere realizzato. L'idea attuale si deve a William

La Direttiva RoHS blocca il tuo lavoro?
rswww.it/rohs: soluzioni per liberarti
dai problemi!

Perché farsi ostacolare dalle nuove Direttive Europee quando c'è RS per aiutarti? Clicca su rswww.it/rohs: trovi subito migliaia di prodotti conformi RoHS, schede tecniche gratuite, informazioni chiare e complete.
Con RS hai sempre le risposte giuste per lavorare con sicurezza.

RoHS Direttiva RoHS: ti basta sapere che c'è RS.

RS

Hewlett (co-fondatore della Hewlett Packard), che lo presentò nella sua tesi di laurea alla Stanford University. Non per caso il circuito, in grado di produrre un segnale ottimamente sinusoidale, fu utilizzato per la prima volta nell'oscillatore di precisione HP 200A.
La limitata distorsione dell'oscillatore di Hewlett risiedeva nella stabilizzazione dell'ampiezza. L'ampiezza del segnale in un oscillatore tende ad aumentare fino alla saturazione.
Hewlett, invece, impiegò una lampada ad incandescenza per limitare (in modo non lineare) il guadagno dell'amplificatore. La resistenza del filamento aumenta infatti con l'aumento della temperatura, a sua volta aumentata dalla corrente per effetto Joule. Il circuito è tale che un aumento della resistenza prodotto da un aumento del livello del segnale, comporti una riduzione del guadagno. Attualmente, per lo stesso scopo, si usano FET o fotocellule, ottenendo distorsioni inferiori allo 0,0008%

Lo schema dell'oscillatore a ponte di Wien è riportato in figura 15. La tensione di squilibrio del ponte viene posta in ingresso ad un amplificatore differenziale (figura 16). Il circuito é composto da un amplificatore operazionale con due tipi di reazione: una *negativa* con R_2 e R_1 come un normale amplificatore ed una *positiva* mediante il ramo del ponte che contiene elementi reattivi. Un simile amplificatore, con una parte di reazione positiva, diventa un oscillatore.

La rete RC, con la sua reazione positiva, riporta una parte della tensione d'uscita all'ingresso del amplificatore che, per effetto della reazione negativa, è di tipo non invertente e quindi non introduce alcuno sfasamento. Per soddisfare la condizione di Barkhausen occorre allora che anche la rete di reazione non introduca alcuno sfasamento.
Il criterio di Barkhausen per l'oscillazione fornisce la condizione (tralasciamo i conti)

$$R_1 > 2R_2$$

e la frequenza dell'oscillazione è:

$$f_0 = \frac{1}{2\pi RC}$$

Attenzione: nelle figure 15 e 16 si ha $R=R_3$ e $C=C_3$.

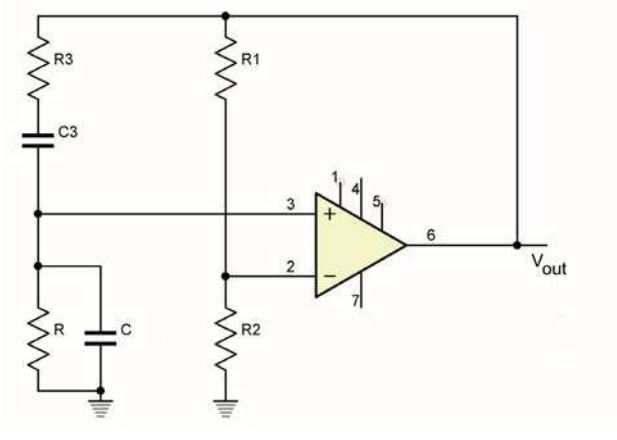


Figura 15 Oscillatore a ponte di Wien

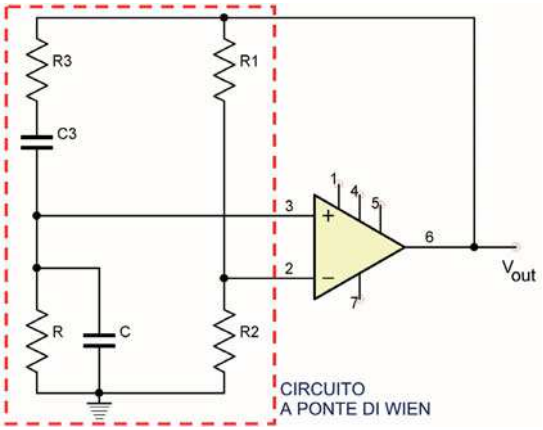


Figura 16 Oscillatore a ponte di Wien

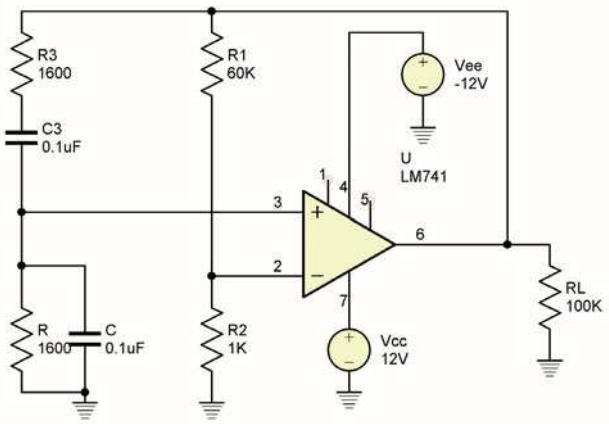


Figura 17 Circuito completo dell'oscillatore a ponte di Wien

Con $R=R_3=1600\ \Omega$, il simulatore fornisce $f_0=11.17$ kHz.
Per una migliore stabilità si può sostituire R_1 con un termistore di tipo PTC in modo che, se l'uscita tende ad aumentare, R_1 si scalda di più e la sua resistenza cresce, aumentando così la tensione di reazione negativa che farà amplificare

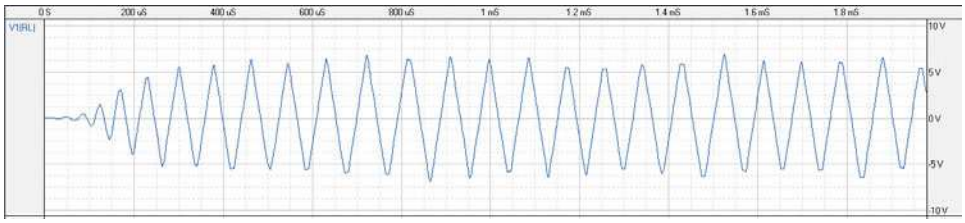


Figura 18 *Simulazione dell'oscillatore a ponte di Wien di figura 17 con C=0.1 μF ed R=R₃=1600Ω*

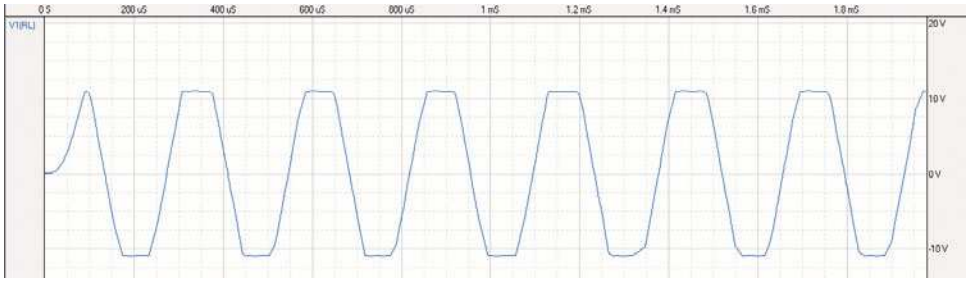


Figura 19 *Simulazione dell'oscillatore a ponte di Wien con C=0.1 μF ed R=R₃=20kΩ*

di meno, tanto quanto basta per stabilizzare l'uscita. Un semplice termistore PTC può essere una comune lampada a filamento del tipo 12V 30/60 mA.

Nella simulazione di figura 19 si è aumentato il valore delle resistenze R ed R_3, diminuendo così la frequenza dell'oscillazione risultante; in tal caso, tramite i cursori di Cadlogix, si legge f_0=3.65 kHz.
Suggeriamo ai Lettori di verificare cosa accade cambiando il tipo di operazionale. Provate ad inserire un amplificatore più veloce per migliorare la resa in alta frequenza.

PROSSIMA PUNTATA

Cari Lettori, il tempo è volato e siamo giunti all'ultima puntata del Corso. Nell'ultima parte introdurremo due strumenti fondamentali per il laboratorio: il multimetro e l'oscilloscopio. Il valido simulatore Cadlogix aiuta nel progetto dei circuiti, ma questi due strumenti saranno per sempre insostituibili. Sfrutteremo parte dello spazio per riassumere alcuni concetti e formule pratiche di uso frequente.

BIBLIOGRAFIA

Un ampio spazio agli amplificatori con retroazione e agli oscillatori è dedicato in:

1. M. Gasparini, D. Mirri
   "**Dispositivi e circuiti elettronici**", Vol. II, Ed. Calderini, Bologna, 1984

e due capitoli ben fatti si trovano in

2. J Millman, C. C. Halkias
   "**Microelettronica**", Ed. Boringhieri, Torino, 1978

Un capitolo più pratico anche se meno didattico si trova nel solito:

3. P. Horowitz, W. Hill – "**The Art of Electronics**", Cambridge University Press, II ed., 1989

ed in inglese una buona trattazione è contenuta in:

4. J Millman, A. Grabel – " **Microelectronics**", McGraw-Hill International Ed., II Ed., 1987

Una trattazione rigorosa della trasformata di Laplace si trova in:

5. M. Marini – "**Metodi matematici per lo studio delle reti elettriche**", Ed. CEDAM, 1999, Padova

ed un capitolo sintetico con numerosi esempi è contenuto in

6. E. Borchi, R. Nicoletti
   "**Circuiti elettrici lineari**", Vol. II "**Esercizi e complementi**", Soc. Ed. Esculapio, 1999, Bologna

La tecnica DDS

Continua la rassegna di progetti basati sulla tecnica DDS. In questo articolo presentiamo un generatore di segnali (sinusoidale, quadra, dente di sega, triangolare, noise e custom) per i quali è possibile impostare la frequenza con la precisione tipica del DDS.
Molti i campi d'applicazione (generatori di frequenza da laboratorio, di riferimento per PLL, DTMF, QRP e in tutte quelle applicazioni dove è richiesto generare segnali a frequenze stabili con notevole precisione.

Oggi giorno nei laboratori tecnici è richiesta la presenza di generatori di segnali con elevate caratteristiche di precisione in frequenza e forma d'onda. La tecnica DDS, ad oggi, permette la generazione di segnali con elevata precisione nella forma e frequenza.
Oggetto di questo articolo è un generatore di segnali, realizzato mediante un microcontrollore PIC, in cui è possibile cambiare la forma d'onda in uscita (si possono scegliere 6 forme diverse) e la frequenza a step prestabiliti. Molteplici possono essere le applicazioni di tale progetto: generatori di frequenza da laboratorio, di riferimento per PLL, DTMF, QRP e in tutte quelle applicazioni dove è richiesto generare segnali a frequenze stabili con notevole precisione.

SOLUZIONI CIRCUITALI

Nei progetti precedenti sulla tecnica DDS, è stata fatta una scelta di base: la forma d'onda dei segnali in uscita era un'onda quadra. Tale scelta era imposta dal fatto che era sfruttata la naturale mappatura dell'onda quadra considerando il bit MSB dell'accumulatore (per ulteriori chiarimenti rimandiamo ai progetti precedenti). Come è stato detto nel tutorial relativo al DDS (Fare Elettronica 240 – Giugno 2005) la generazione mediante DDS richiede una ROM su cui mappare la forma d'onda da generare e un DAC che generi la forma d'onda analogica.
Nel presente progetto è richiesta la possibilità di generare dei segnali che non siano un'onda quadra per questo sono necessari una ROM e un DAC. Vediamo quali accorgimenti tecnici sono stati usati per implementare la ROM e il DAC tenendo conto che il progetto si basa esclusivamente su un microcontrollore PIC.

Mappatura del segnale

Per la mappatura delle forme d'onda si può ricorrere alla memorizzazione di queste nello spazio di memoria del microcontrollore riservato al codice (program words memory) e utilizzando l'istruzione RETLW. Un'altra soluzione, laddove è possibile, è di implementare la funzione matematica che genera i campioni del segnale in uscita in funzione dell'ingresso fornito dall'accumulatore. L'unica cosa da fare, per la prima soluzione, è scrivere il relativo codice asm e ciò può essere fatto semplicemente avvalendosi di un foglio di calcolo tenendo conto del numero di punti da mappare come funzione dei **N** bit dell'accumulatore e il numero dei bit con cui quantizzare la forma d'onda in uscita. Nel caso della seconda soluzione è necessario individuare la funzione matematica.

Il PWM come DAC

Per quanto riguarda il DAC una soluzione sareb-

Generatore di segnali a frequenza variabile

di Salvatore Torrisi
microst@microst.it

be quella di collegarne uno, a resistori pesati ad una porta del PIC con conseguente appesantimento dell'hardware. La soluzione adottata nel progetto proposto si basa sulla generazione di una tensione da un'onda quadra modulata in PWM (Pulse Wave Modulation).
Indicando con **D** il duty cycle dell'onda quadra, inteso come il rapporto tra il tempo in cui l'onda è al valore alto e il suo periodo, la componente continua dell'onda PWM è pari a: **Vc=Vmax * D**.
Si può pensare di realizzare un DAC con un generatore PWM dove l'ingresso è il duty cycle e l'uscita è la componente continua ottenuta mediante un filtro LPF.
Quindi il singolo campione del segnale da generare può essere convertito in analogico da un segnale ad onda quadra con un duty cycle D il cui valore è definito dalla mappatura del segnale.
Compatibilmente all'utilizzo di un microcontrollore, per la generazione del segnale PWM, si può usare il modulo PWM presente in alcuni microcontrollori. Quello che abbiamo scelto per questo progetto è il PIC16F628 che è dotato di in modulo PWM a 10 bit massimi (bit con cui è definito il duty cyle).

Dimensionamento dei parametri

Per rendere operativa l'idea di realizzare un DAC con il modulo PWM bisogna dimensionare alcuni parametri tipici del PWM, del DDS e quelli provenienti dalla loro implementazione trovando le relazioni che li legano.
I parametri del PWM sono: **ND** (numero di bit per definire il duty cycle) e **fpwm** (frequenza del segnale pwm). I parametri DDS sono: **N** e **fclk** rispettivamente numero di bit e frequenza di generazione.
Altri parametri sono: **Ncyle** (numero di cicli macchina per un ciclo di generazione DDS), **Xtal** (frequenza del quarzo utilizzato per il micro), **Np** (numero d'onde pwm per singolo campione), **Ndac** (numero bit del DAC/PWM).
La relazione che lega **ND**, **fpwm** e **Xtal** è definita dalle specifiche del modulo PWM. (vedi il datasheet del PIC16F628).

Essendo **fclk = Xtal / (4 Ncycle)** e **Np = fpwm / fclk** si ha che:
Ncycle=Np*Xtal / (4*fpwm) [1]
oppure
Np = 4*Ncycle* fpwm / Xtal [2]

Si può quindi fissare Np per cui il valore di Ncycle scaturisce dalla (1) quindi anche il valore di fclk e i valore massimo di frequenza **fo_max=fclk / 2**. In tal caso sì da maggiore peso alla specifica risoluzione poiché maggiore è il valore di *Np* maggiore è la qualità del segnale in uscita dal filtro LPF a spese del valore massimo della frequenza ottenibile in uscita e del minimo incremento di frequenza **Δf=fclk / 2^N**
Se si vuole massimizzare i valore della frequenza in uscita a spese della risoluzione si fissa fclk quindi dalla relazione (2) si ricava *Np*.
In questa sede si vuole dare maggiore peso alla risoluzione, quindi fissato per Xtal=20Mhz e ND=8, Ndac=8, N=16, Np=3, si ha:

fpwm=78120 Hz

e dalla relazione 1

Ncycle = 3*20e6 / (4*78120) = 193

Quindi il numero di cicli macchina necessari per un ciclo di generazione DDS deve essere 193 per avere al tempo stesso 3 cicli pwm completi.
Di contro **fclk, fo_max** e **Δf** saranno:

fclk = Xtal / (4* Ncycle) = 20e6 / (4*193) =
= 25.906 kHz
fo_max = fclk/2 = 19.53/2 = 12.953 kHz
Δf=fclk / 2^N= 19530 / 2^16 = .395 Hz

Rifacendo i calcoli a ritroso in modo da avere **Δf=0.4 Hz** si ha **Ncycle =191, fclk=26178 Hz** e **Np=3.0**

Infine avendo fissato N=8 il numero di punti in cui dovranno essere mappate, le forme d'onda sono pari a 2^8=256 mentre, essendo ND=8, la quantizzazione della dinamica del segnale è fatta in 256 livelli (Vmax-Vmin = 5V).

CARATTERISTICHE DEL GENERATORE

Sulla base delle considerazioni teoriche fatte nei paragrafi precedenti con il generatore di segnale a frequenza variabile basato sulla tecnica DDS sarà possibile:

1. Scegliere 3 forme diverse di segnale canonici: sinusoidale, quadra, triangolare, dente di sega, noise.
2. Mappare una forma d'onda (custom) definita direttamente dall'utente.
3. Cambiare incrementare/diminuire la frequenza del segnale scelto.
4. Impostare lo step di variazione di frequenza.

Per quanto riguarda i punti 1 e 2, le sei forme d'onda sono gia pre-memorizzate nel file.hex fornito con il circuito. Mediante il tasto **SEL** è possibile scegliere la forma d'onda. Per mappare la forma d'onda custom (punto 2) bisogna intervenire direttamente sul file *sigen.hex* inserendo da una data locazione i valori dei campioni. Per aumentare/diminuire la frequenza bisogna agire sui tasti **UP/DOWN** mentre il valore dello step di variazione è definito direttamente da codice intervenendo su una data locazione del file sigen.hex.

Caratteristiche dei segnali generati

Di seguito un quadro riassuntivo delle forme d'onda in uscita e alcune loro caratteristiche elettriche e d'implementazione.

Sinusoide: Vpp=5V Vm=2,5V, mappata nel codice.
Quadra: Vpp=5V Duty Cycle=50%funzione matematica.
Dente di sega: vmax = 5V per t = T, funzione matematica.
Triangolare: simmetrica Vmax = 5V per t =T/2, funzione matematica.
Noise: rumore bianco, mappata nel codice
Custom: definita dall'utente, mappata nel codice.

Caratteristiche tecniche del circuito

Per Xtal=20Mhz e ND=8, Ndac=8, N=8, Np=4 si ha:

- Tensione d'alimentazione: 2.7 - 5V
- Range Frequenza d'uscita: 0 - 13 KHz (1000 Hz iniziale)
- Frequenza iniziale: 1 KHz
- Range Split: 0 - 10 KHz
- Split (default): 100 Hz
- Tensione d'uscita: 0 - 5V

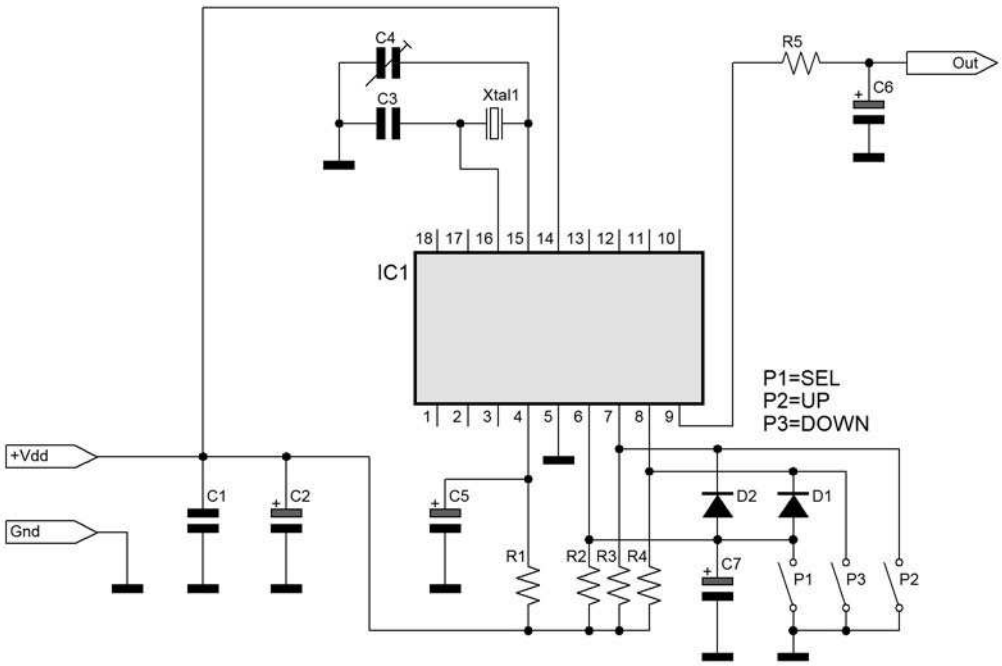


Figura 1 Schema elettrico del generatore di segnali a frequenza variabile

Altre caratteristiche elettriche sono quelle riportate sul datasheet del microcontrollore utilizzato.

Descrizione del progetto

In figura 1 è riportato lo schema elettrico del generatore di segnale. Il circuito è costituito da un microcontrollore PIC16F628 e da alcuni componenti passivi. Per la sua realizzazione possono essere usati altri modelli della famiglia PIC16X con modulo CCP.

Funzionamento del circuito

Il PIC16F628 implementa un accumulatore di fase a **N=8** bit. Ad ogni ciclo di generazione DDS, l'uscita dell'accumulatore indirizza la parte di programma destinata come look-up table dove è mappata la forma d'onda, selezionata mediante il tasto SEL. Il valore di ritorno della routine è caricato nei registri di configurazione del duty cycle del modulo CCP in configurazione PWM. In uscita saranno generati *Np* onde quadre con duty precedentemente fissato.
Premendo il tasto UP/DOWN, la frequenza aumenterà/diminuirà di una quantità **ΔFo** impostabile all'interno del programma e dipendente dal valore di *XTAL*.
In dettaglio, quando il tasto UP (o DOWN) è premuto, il pin RB1 (RB2) della PORTB è forzato a GND essendo questo normalmente a Vdd.
Contemporaneamente, in virtù della presenza del diodo D1 (o D2), il pin RBO è messo a massa scatenando un interrupt impostato con il fronte di discesa di RB0. Il diodo D1 (o D2) è connesso in modo da far passare il fronte di discesa del pin RB1 (o RB2) su RB0.
Internamente s'incrementa (o decrementa) la variabile che definisce il valore della frequenza in uscita e ripartirà la generazione con il nuovo valore di frequenza.

Elenco componenti	
Sigla	**Valore**
R1, R2, R3, R4	10 KΩ 1/4 W
R5	Vedi testo
C1, C7	100 nF poliestere
C2	100 µF 16 V elettrolitico
C3	5 pF ceramico
C4	4-20 pF compensatore
C5	1 µF 16 V elettrolitico
C6	Vedi testo
D1, D2	1N4148
P1, P2, P3	Pulsanti Normalmente Aperti
IC1	PIC16F628
XTAL	20 MHz (*)

(*) La scelta del valore del quarzo è estremamente importante poiché il valore della frequenza *XTAL* rientra nel calcolo d'alcuni parametri. Se si usano valori diversi per il quarzo Xtal si avranno valori di frequenza e di step diversi da quelli per Xtal =20MHz.

Scelta e dimensionamento del filtro LPF

La qualità del filtro LPF da porre in uscita al circuito è importante perchè definisce la qualità del segnale. In questa sede è stato scelto un filtro LPF RC formato da R5 e C6 ma è possibile utilizzare filtri d'ordine maggiore sia passivi sia attivi. Per XTAL=20 MHz e Ncycle=191 si ha una frequenza massima di:

$$fo_max = fclk/2 = 19.53/2 = 12.953\ kHz$$

Posto R5=1 KΩ per avere una frequenza di taglio pari alla massima frequenza d'uscita C6 deve avere il valore definito secondo la formula:

$$fo_max = fclk\ /\ 2 = [Xtal\ /\ (4^*\ Ncycle)]\ /\ /2=1/(2^*\pi^*R5^*C6)$$

da cui

$$C6=4\ *\ Ncycle\ /\ (\pi^*R5^*\ Xtal)$$

$$C6=4^*191/(3.14^*1000^*20e6)=\ 12.165nF$$

Si può scegliere per C6 il valore di 15nF. La presenza del filtro LPF in uscita comporta una deformazione dei fronti di salita e discesa dell'onda quadra. Si può pensare di sconnettere il capacitore del filtro quando si genera in uscita l'onda quadra.

Montaggio

In figura 2 e 3 sono riportati il circuito stampato lato rame e lato componenti del generatore. Il circuito può essere parte integrante di progetti più grandi e complicati, quindi, il circuito stampato è utilizzabile per la realizzazione del solo generatore.
Tutti coloro i quali hanno realizzato e/o visionato il progetto relativo al riproduttore di suonerie noteranno che lo schema elettrico del generato-

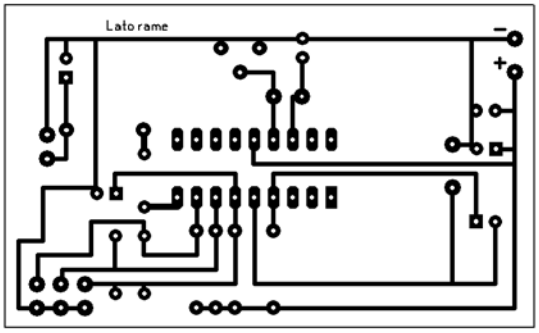


Figura 2 Circuito stampato (lato rame) in scala 1:1

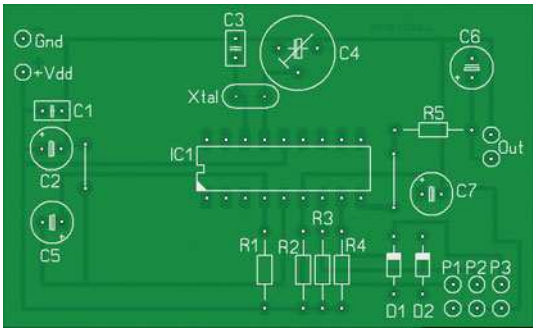


Figura 3 Piano di montaggio

re è praticamente uguale a quello del riproduttore. L'unica differenza sta nel pin d'uscita che, nel caso del riproduttore, è il RB7 (pin 13 del PIC) mentre nel generatore è il RB3 (pin 9) e nell'assenza dei diodi led. Questo suggerisce di poter riutilizzare il pcb del riproduttore realizzando un ponticello tra il pin 9 e il pin 13, essendo questo disattivo per il progetto del generatore.

Taratura e Collaudo

La taratura del circuito deve essere fatta in due fasi: taratura in frequenza e controllo delle forme d'onda. Per la prima fase è necessario utilizzare un frequenzimetro. Dopo aver collegato il probe del frequenzimetro sul pin 15 del PIC, mediante un cacciavite per RF, bisogna agire su C4 ruotandolo in modo da leggere la frequenza del quarzo utilizzato più vicina al suo valore nominale. Se per qualche motivo non si riesca ad ottenere il valore del quarzo, si può cambiare il valorie di C3 o C4. Per verificare l'esattezza dello SPLIT, premendo UP/DOWN dovreste trovare un valore maggiore/minore di frequenza rispetto a quel letto prima del pressione sul tasto.
Per la taratura della forma d'onda bisogna collegare un oscilloscopio al pin d'uscita del circuito e visualizzare le diverse forme d'onda per verificarne ampiezza, forma e frequenza. Vista la bassa frequenza di lavoro è possibile utilizzare uno dei tanti oscilloscopi realizzati per PC con scheda audio, avendo cura di interporre tra il probe dell'oscilloscopio e l'uscita del circuito un buffer di tensione (realizzato per esempio con uno stadio del LM324 o simili).

LA FORMA D'ONDA "CUSTOM" E LE COSTANTI DEL CIRCUITO

Come detto prima è possibile caricare una forma d'onda custom ovvero definita dall'utente. Per esempio si può pensare di generare un segnale con andamento random (per esempio rumore bianco, oppure segnali impulsivi), segnali video, segnali di trigger e così via. La mappatura di questo segnale è fatta nel codice in 256 punti, con quantizzazione dei campioni a 8 bit. Per inserire i valori relativi alla forma custom bisogna intervenire direttamente sul file **sigen.hex** relativo al progetto e cambiare i valori che vanno dalla locazione 500h a 5FFh mediante un programma che permette di editare il file hex. Il valore da inserire nel file hex è 34xx dove xx è il valore in esadecimale del campione del segnale. Per default, come forma d'onda custom, è mappato un segnale costante alla tensione di 2,5 V.

Definizioni delle costanti

Per funzionamento del circuito è necessario definire delle costanti che possono essere cambiate intervenendo sul file **sigen.hex** in determinate locazioni. Le costanti da definire sono SPLIT e INI che definiscono rispettivamente lo step di incremento/decremento della frequenza per ogni operazione di UP/DOWN e la frequenza del segnale allo start-up del circuito. Le locazioni relative alla costante INI sono: 77h e 79h. Il valore da inserire è: 30xx dove xx è il valore esadecimale della costante. (77h per la MSb e 79h per la LSb di INI).
Le locazioni relative alla costante SPLIT sono: 7Bh e 7Dh. Il valore da mettere è: 30xx dove xx è il valore esadecimale della costante. (7Bh per la MSb e 7Dh per la LSb di SPLIT). Nel riquadro "Esempio della definizione di INI e SPLIT" è mostrato come calcolare i valori di INI e SPLIT per una frequenza iniziale di 1000 Hz e uno split di 100 Hz (valori di default).

Esempio della definizione di INI e SPLIT

Fissati XTAL=20 MHz, N=8, Ncycle=191 => CLK = 25.906 KHz, STEP=0.4 Hz, vogliamo che il valore iniziale di Fo sia 1 KHz e ΔFo=100 Hz ed utilizzando le formule della prima sezione si ha:

1. Fo = INI *STEP =>1000 = FCW *.4
2. INI = Fo / STEP = 1000 / .4 = 2500
3. Convertiamo in hex INI = 9C4
4. Scrivere 3009 nella locazione 77h
5. Scrivere 30C4 nella locazione 79h

Per il calcolo di SPLIT si procede così:

1. ΔFo = SPLIT *STEP =>100 = SPLIT *.4;
2. SPLIT = 100 / .4 = 250
3. Convertiamo in hex SPLIT = FAh =>
4. Scrivere 3000 nella locazione 7Bh
5. Scrivere 30FA nella locazione 7Dh

APPLICAZIONI E MIGLIORAMENTI

Lo scopo di questo circuito è di supportare la teoria relativa al DDS. Le caratteristiche dei segnali in uscita non sono quelle di un generatore professionale. Se si vuole migliorare il progetto si può pensare di aggiungere tutte quelle funzioni che un buon generatore di segnali può avere. Questo circuito può essere quindi la base di progetti più complessi, in cui si può inserire la visualizzazione della frequenza generata, cambiare lo split, memorizzare valori, sostituire i pulsanti con un rotary encoder, trigger, eccetera.

CONCLUSIONI

Con questo progetto, come con i precedenti di questa serie, si è voluto ulteriormente dimostrare le potenzialità della tecnica DDS. L'utilizzo di tale tecnica con microcontrollori che dispongono di buone risorse hardware, permette di ottenere buoni risultati con costi, in termini di sviluppo software e hardware, abbastanza modesti. Vi ricordo che è possibile scaricare il codice sorgente dal sito di Fare Elettronica.

Introduzione ai

I DSP sono processori disegnati per risolvere una particolare classe di problemi computazionali. Una delle applicazioni più semplici a cui si pensa quando si parla dei DSP è l'elaborazione numerica dei segnali, finalità in rapida crescita in molteplici aree di applicazione come: comunicazioni wireless, elaborazioni audio e video, e controlli automatici.
A causa dell'enorme successo di queste realizzazioni, la varietà di tali processori si è espansa notevolmente. Abbiamo deciso con questo speciale di allargare gli orizzonti circa questi dispositivi, parlandone in generale e con un esempio "di successo" come la famiglia DSP TMS320 C6000 della Texas Instruments.

Oggi i DSP sono sofisticati processori special-purpose con grandi capacità che possono implementare diversi tipi di architetture integrate.
La caratteristica base che accomuna tali processori è un'operazione particolare disegnata per realizzare task ripetitivi, numericamente impegnativi ad alte prestazioni. L'operazione comune a tutti i DSPs è definita attraverso la sigla M.A.C., dove l'acronimo sta per "Multiply and Accumulate", un'operazione che possa essere fatta una o più volte in un singolo ciclo di istruzione. L'istruzione MAC, come è facile capire, realizza insieme due operazioni matematiche: la somma ed il prodotto, per esempio A=A + BxC. Alcuni tipi di applicazione ben supportati da questo tipo di processore sono:

- **Le serie di Taylor**. Nel 1712, il matematico *Brook Taylor* definì una formula che potesse approssimare una funzione. Presa una funzione f(x) (in alcune restrittive ipotesi di continuità), è possibile approssimarla in un intorno di x=a con la seguente espressione, dove figura $f^n(x)$ ad indicare la derivata ennesima di f(x):
  $f(x) = f(a) + f^1(a)(x-a) + f^2(a)(x-a)^2/2! + \ldots + f^n(a)(x-a)^n/n!$
  più un resto che lasciamo ai testi di analisi.

  Questa espressione realizza, nelle varie discipline, un modo standard per operare l'approssimazione di funzioni altrimenti non valutabili. Si può notare come ogni termine di tale relazione è un'operazione di moltiplicazione ed accumulo e generalmente il resto risulta essere sempre più trascurabile, quanto più ci avviciniamo alla reale funzione approssimata e cioè quanti più termini è possibile valutare.
- **I filtri FIR (Finite Impulse Response)**. Sono usati per filtrare un flusso continuo di informazioni rappresentante audio o video. Un esempio di applicazione può essere la ricezione di un segnale audio in presenza di rumore sovrapposto a frequenza di 1 KHz, tale segnale può essere ripulito da disturbi attraverso un filtro FIR. Supponendo di voler filtrare il segnale x(t) per produrre il segnale y(t), la formula generalizzata di solito applicata è:
  $y(t) = h_0x(t) + h_1x(t-1) + h_2x(t-2) + \ldots + h_nx(t-n)$
  dove i vari h_i sono i coefficienti del filtro calcolati secondo opportune metodologie.
- **La trasformata di Fourier**. La trasformata di Fourier fu sviluppata da *Joseph Fourier* nel

processori DSP

di Enrico Raffone
e.raffone@farelettronica.com

corso dei primi anni del '800. È un metodo per poter rappresentare ogni tipo di funzione con una ben determinata approssimazione, attraverso la sovrapposizione di una serie di funzioni sinusoidali pure. La trasformata di Fourier ha diverse varianti, tra cui la *Fast Fourier Trasform* (FFT) e la *Discrete Cosine Transform* (DCT), queste trasformate sono utilizzate di solito per rimuovere rumore o filtrare frequenze indesiderate, mentre l'inversa della trasformata di Fourier viene spesso usata in applicazioni in cui si ricostruiscono immagini così da renderle più "chiare" per specifiche applicazioni.

I più recenti DSP hanno due o più unità MAC, riuscendo così a realizzare più operazioni in parallelo. La caratteristica degli algoritmi di elaborazione numerica di avere in se un intrinseco parallelismo di operazioni, ha facilitato lo sviluppo di architetture *VLIW* (Very Long Instruction word) nei DSP (vedi approfondimento).

Possiamo individuare in figura 2, alla luce delle considerazioni fatte nella finestra di approfondimento, una tipica architettura VLIW, con 4 diverse unità funzionali.
Altra interessante caratteristica comune a tale classe di processori è la possibilità di completare diversi accessi in memoria in una singola istruzione. Questo permette di realizzare, in contemporanea, il caricamento (FETCH) di una nuova istruzione e il caricamento degli operandi e/o il salvataggio del risultato dell'istruzione precedente in memoria. Per esempio per realizzare un prodotto in un filtro FIR, la maggior parte dei DSP sono capaci di realizzare una operazione MAC ed in contemporanea caricare il valore del coefficiente e quello del campione di segnale per l'operazione successiva. Questi accessi multipli in memoria sono comunque sottoposti ad alcune restrizioni, per esempio una delle locazioni di memoria accedute deve risiedere on-chip, oppure tali accessi multipli possono essere realizzati solo con certe istruzioni.

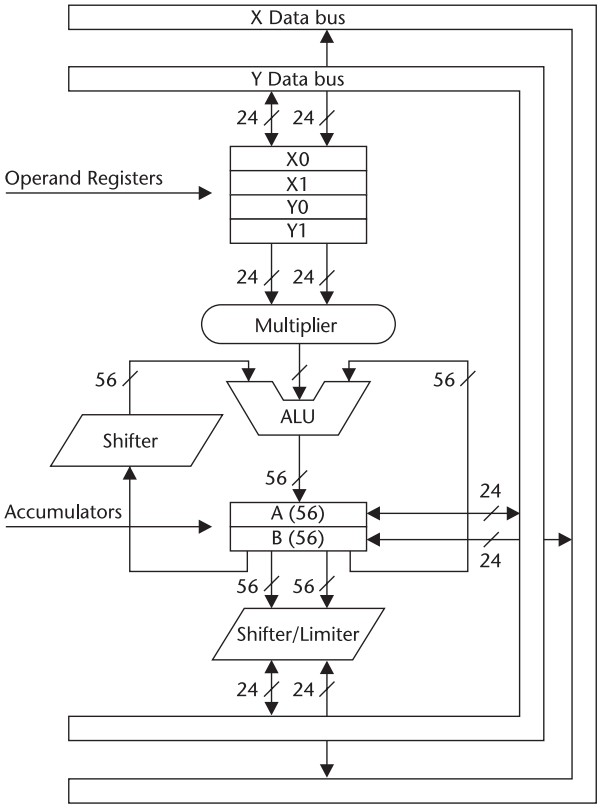


Figura 1 *Rappresentazione del data path di un processore DSP (Motorola DSP560xx)*

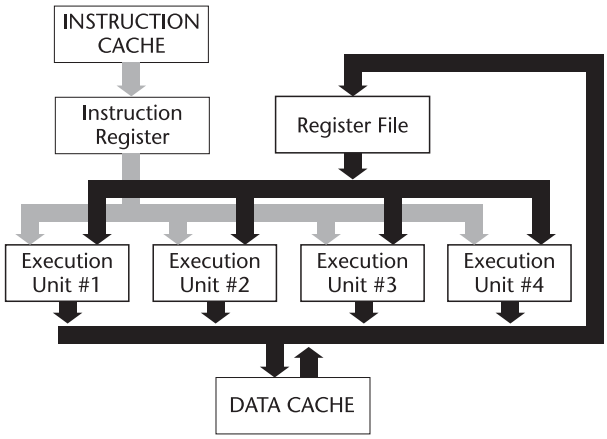


Figura 2 *Architettura VLIW*

Per facilitare accessi multipli in memoria, spesso i DSP hanno più bus on-chip, memorie multiporte (figura 3) ed in qualche caso banchi di memoria indipendenti.

Una terza caratteristica per migliorare i DSP nelle elaborazioni aritmetiche è la presenza di un'unità di generazione indirizzi. Una volta che tutti i registri di indirizzamento sono stati confi-

LE ARCHITETTURE VLIW

VLIW (Very Long Instruction Word). Con tali architetture si implementa un parallelismo interno al processore, ottenuto mediante duplicazione delle unità funzionali, che lavorano in contemporanea. Invece di avere un serie di n istruzioni in sequenza, abbiamo l'impacchettamento di n istruzioni insieme in un'unica macroistruzione:

I1	I2	I3	…	In
↓	↓	↓	↓	↓
U	U	U	…	U

Ogni istruzione viene eseguita da una singola unità funzionale. Le architetture VLIW di sicuro non sono implementate su macchine general-purpose, in quando già con due pipe in parallelo (architetture superscalari) c'è da far fronte ad una serie di problemi che lasciamo ai testi di Architetture degli Elaboratori. I VLIW sono architetture caratteristiche di macchine studiate per risolvere una particolare classe di problemi computazionali. La programmazione non viene effettuata direttamente dall'utente, ma esistono delle librerie già scritte così da eliminare qualsiasi tipo di conflitto.
Le istruzioni del programma sono *impacchettate* dal compilatore in modo da utilizzare tutte le diverse stazioni (unità di elaborazione). Si cerca di alimentare contemporaneamente le diverse stazioni con operazioni di addizione, di moltiplicazione ecc.
Dunque, un certo numero di istruzioni vengono eseguite in parallelo da unità disgiunte. Occorre però essere in grado di creare la macroistruzione a partire da microistruzioni, in base alle unità funzionali disponibili.
Quindi si delinea una serie di problemi: se c'è una sola addizione da fare non è possibile ovviamente inserirne due nella relativa stazione; se due operazioni, dello stesso o di diverso tipo, vanno in conflitto sui dati, non possono essere caricate in un ordine casuale, ecc.
Inoltre, gestire le istruzioni di salto diventa ancora più difficile, poiché non si possono fare previsioni (come in altre architetture). Se si effettua un salto (condizionato) che non andava eseguito (quindi sbagliando) non c'è modo di tornare indietro. Se per esempio abbiamo a disposizione 8 unità funzionali, abbiamo quindi impacchettato 8 istruzioni in un'unica macroistruzione e, nella fase di esecuzione, ci si accorge che l'istruzione I_2 non doveva essere eseguita perché l'istruzione precedente è un salto, allora la situazione è difficile da gestire perché abbiamo impacchettato già I_2-I_7, che potrebbero già aver modificato lo stato del processore.
Dove trovano applicazione queste architetture che, in apparenza, non fanno altro che introdurre problemi notevoli? Non nei comuni processori general-purpose, ma piuttosto nei **processori DSP (Digital Signal Processing)**. Un DSP è un processore orientato all'elaborazione dei segnali; alcuni DSP sono estremamente verticalizzati (special-purpose) in tale finalità, altri sono un po' più general-purpose.
Gli utenti DSP generalmente non *programmano* direttamente su tali processori, ma li usano con librerie già pronte, perché l'elaborazione numerica dei segnali si basa sul calcolo di trasformate di Fourier e di altro genere. Una libreria per DSP con architettura VLIW è ideata con l'obiettivo di risolvere non uno, ma *n* problemi. Ciascun problema è risolto attraverso l'utilizzo di una istruzione della libreria, costituita da istruzioni elementari che sono state *impacchettate nel modo più conveniente*. Quindi una libreria offre le *migliori implementazioni* delle varie operazioni, sfruttando al meglio il parallelismo dei blocchi funzionali.

pcb.it

forum italiano delle produzioni elettroniche

La guida on-line chiara e razionale per chi opera nella produzione e nella subfornitura elettronica, dalla progettazione all'assemblaggio

inserimento gratuito

- assemblaggio
- attrezzature e macchine
- circuiti stampati
- strumentazione
- progettazione
- cablaggi
- servizi

Un motore di ricerca dedicato per trovare nomi, indirizzi e attività di oltre 2000 aziende operanti sul territorio nazionale

Un punto di riferimento e informazione per gli addetti ai lavori su eventi, manifestazioni, workshop, e nuovi prodotti

Utili link ad associazioni, istituzioni, università e pubblicazioni del settore italiane ed estere

ELETTRIMPEX

Promosso e gestito dal **Consorzio Elettrimpex**
Per informazioni: tel. +39 02 210.111.244 • fax +39 02 210.111.222 • e-mail: info@pcb.it

POWERED BY BLWEB http://www.blweb.it

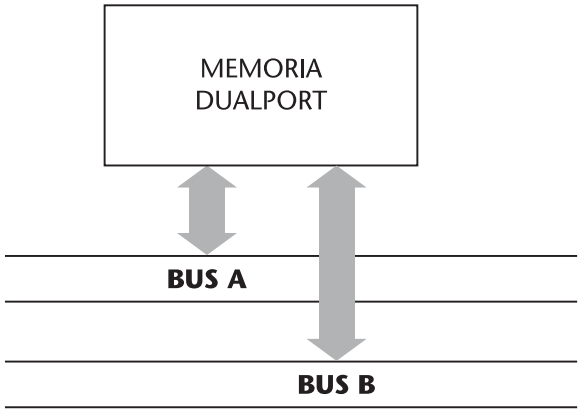


Figura 3 Memoria Multiporte

gurati, l'unità di generazione indirizzi lavora in background, cioè senza usare l'insieme dei registri principali del processore. Questa unità realizza gli indirizzi richiesti per caricare gli operandi in parallelo, con l'esecuzione di istruzioni aritmetiche. Bisogna sottolineare che i processori general-purpose sovente richiedono cicli extra per generare gli indirizzi necessari per gli operandi. Tale unità di solito prevede più modi di indirizzamento per adattare nel modo migliore il DSP alle diverse applicazioni. Tra i diversi modi di indirizzamento, quello più comune è l'indirizzamento detto *indiretto a registro* con post incremento, questo modo è tipico di situazioni in cui si devono realizzare computazioni ripetitive su dati salvati sequenzialmente in memoria. L'indirizzamento *in modulo* è anche esso spesso supportato per semplificare l'uso di buffer circolari. Altri processori prevedono l'indirizzamento *bit-reserved* che incrementa la velocità di alcuni algoritmi di FFT (Fast Fourier Transform).

Come già specificato capita spesso nelle applicazioni per DSP di imbattersi in cicli e computazioni ripetitive, ciò ha portato alla progettazione di architetture integrate, disegnate per realizzare efficientemente tali *loop*. Spesso sono introdotte speciali istruzioni che permettono ai programmatori di implementare algoritmi ripetitivi senza spendere alcun ciclo di istruzione per aggiornare e testare il contatore o fare branch

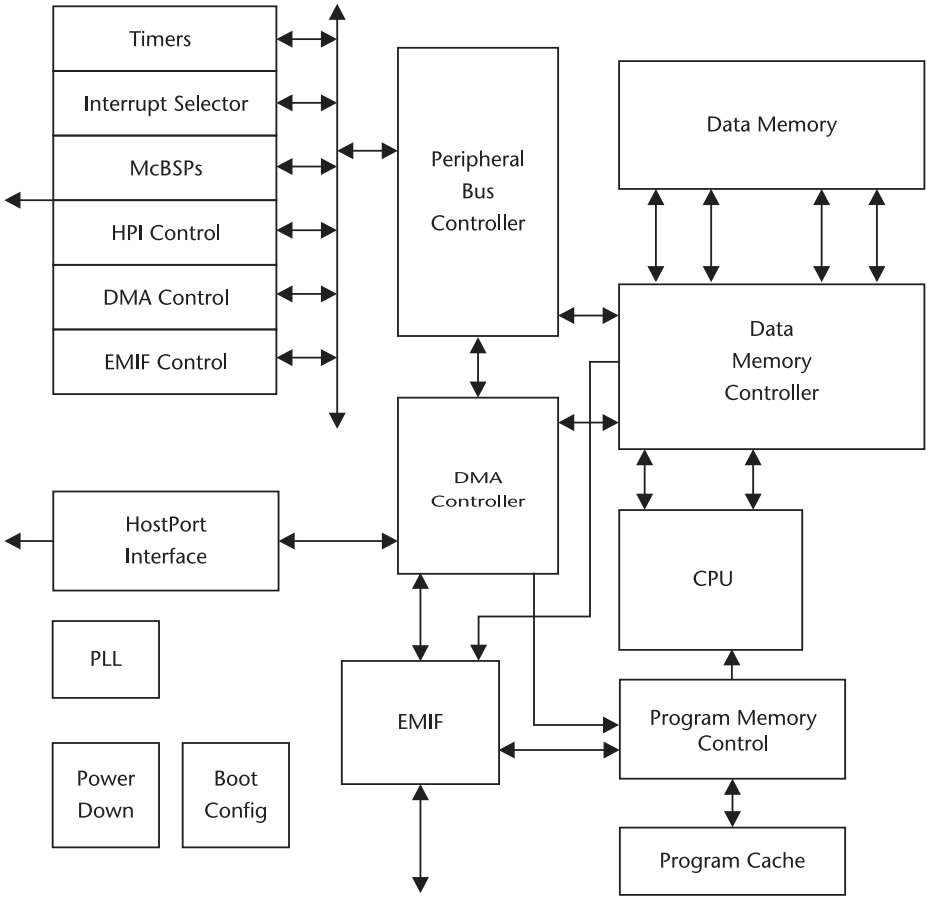


Figura 4 Diagramma funzionale TMS320C6000

all'inizio del loop.
Veniamo quindi ad una tendenza comune anche ad altri tipi di processori; per proporre connettività a basso costo e ad alte prestazioni, molti DSP incorporano più interfacce I/O seriali o parallele e meccanismi specializzati di I/O come interruzioni a basso overhead e direct memory access (DMA) , così da permettere trasferimenti di dati con piccoli, o del tutto assenti, interventi da parte del resto del processore. Di particolare utilità è il DMA che permette di trasferire grosse quantità di dati da una memoria all'altra senza impegnare minimamente il processore, naturalmente il DMA deve essere prima configurato dal DSP.

In figura 5 è possibile vedere l'architettura del Texas Instruments TMS320C6000, che presentiamo nelle caratteristiche salienti alla fine di questo speciale, si nota la presenza di un DMA e di un controllore di memoria EMIF. In realtà nello sviluppo odierno delle architetture non è più corretto parlare semplicemente di processore DSP, si dovrebbe parlare di sistema DSP, perché il processore ormai è solo una parte di una più complessa struttura che permette di processare segnali numerici, inoltre le prestazioni stesse del DSP sono legate alle caratteristiche di tutto quello che è nel sistema.

La crescente popolarità di alcune funzioni di elaborazione numerica come *speech-coding* e *audio-processing* ha portato ad implementare caratteristiche legate ai DSP su processori general-purpose. Oggi tutti i produttori di processori hanno aggiunto capacità di elaborazione numerica ai loro prodotti. Per esempio l'estensione MMX e SSE all'instruction set dei Processori Intel della famiglia Pentium sono esempi di estensioni multimediali che hanno molto a che fare con i DSP.

In alcuni casi, i progettisti di sistema preferiscono utilizzare processori general-purpose piuttosto che DSP. Le architetture dei processori general-purpose necessitano di diverse istruzioni per realizzare operazioni che possono essere realizzate con un'unica istruzione da un processore

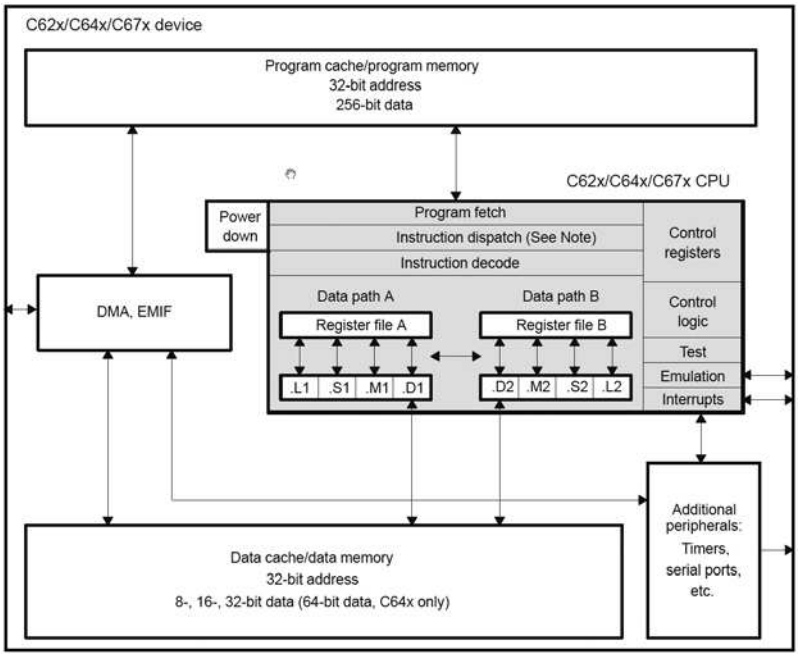


Figura 5 Architettura TMS320C6000

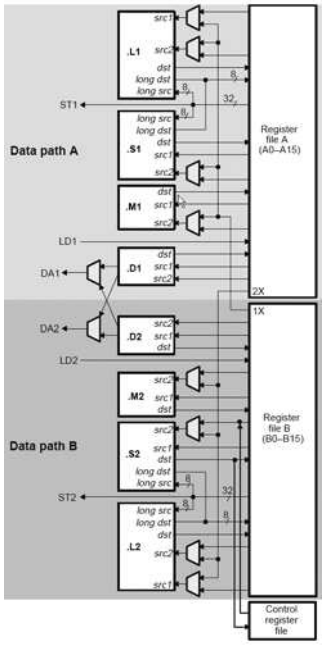


Figura 6 Particolare dei due data path

DSP, d'altro canto molti processori general-purpose lavorano a clock molto elevati. L'utilizzo di processori general-purpose consente di ridurre le dimensioni del sistema e di abbassare i prezzi rispetto all'utilizzo separato di un DSP per realizzare le elaborazioni numeriche. Dall'altronde utilizzando processori general-purpose si complica la parte di programmazione e risulta difficoltoso il mantenimento del codice.

Applicazioni

Dopo aver fatto una larga introduzione sulle architetture, possiamo dire che i DSP trovano uso nei campi di applicazione più disparati, dai sistemi radar all'elettronica di consumo. Sono usati soprattutto nei sistemi embedded di larga produzione, come telefoni cellulari, dischi rigidi e riproduttori portatili di audio e video. In queste applicazioni il costo e l'integrazione hanno una grande importanza, oltre ad un altro fattore critico, il risparmio energetico.
È ovvio che non esiste un processore DSP adatto ad ogni campo di attività, ma bisogna di volta in volta scegliere quello giusto per le proprie esigenze. Per esempio un'altra importante classe di applicazioni riguarda i dispositivi che implementano algoritmi per scopi particolari, quindi a risolvere problemi per cui bisogna studiare una soluzione architetturale specifica, come nel caso dei sonar e degli apparati di monitoraggio sismico. In questo caso la produzione non è su ampia scala e la progettazione è più complessa.

Aritmetica

Altro aspetto per cui si può fare una distinzione tra i vari processori è il tipo nativo di aritmetica usata. La maggior parte dei DSP usa una aritmetica a virgola fissa, i tipi più recenti utilizzano l'aritmetica in virgola mobile, dove ricordiamo i valori sono rappresentati da una mantissa ed un esponente.
L'aritmetica a virgola mobile è più flessibile rispetto quella a virgola fissa. Con la virgola mobile i progettisti di sistema hanno accesso ad un range di valori più ampio. I DSP in virgola mobile sono in genere più facili da programmare, ma hanno un consumo energetico più elevato. Ciò è il risultato di una circuiteria più complessa richiesta da tali processori che impone un utilizzo di un'area di silicio più grande. La facilità d'uso dei processori a virgola mobile è dovuta al fatto che in molti casi il programmatore non deve farsi carico di problemi relativi alla precisione.
Nella maggior parte delle applicazioni a larga diffusione si usa un'aritmetica a virgola fissa perché le priorità sono i costi e il basso consumo. In questo caso i programmatori determinano il range di valori e la precisione richiesta per le loro applicazioni, analiticamente o con simulazioni, aggiungendo poi nel codice operazioni di scaling se necessario. La scelta dell'aritmetica in virgola mobile viene preferita nel caso di applicazioni che hanno requisiti particolari o dove la semplicità di sviluppo è più importante del costo unitario.
È comunque possibile implementare un'aritmetica in virgola mobile su processori che hanno invece un'aritmetica in virgola fissa, usando delle routine software che emulano il comportamento di un DSP a virgola mobile. In ogni caso queste routine sono generalmente molto costose in termini di cicli del processore.

Numero di bit

Quasi tutti i DSP a virgola mobile utilizzano word da 32 bit. Quelli a virgola fissa utilizzano invece word da 16 bit. La grandezza delle word ha un grosso impatto sui costi, questo perché influenza la grandezza del chip, il numero di pin del package ed anche le dimensioni delle eventuali memorie esterne.

UN ESEMPIO DI ARCHITETTURA DSP - TEXAS INSTRUMENTS DSP TMS320 C6000

Introduzione ai DSP TMS320C6000

I DSP TMS320C6000 fanno parte della famiglia TMS320 e comprendono decine di versioni diverse, con diverse frequenze di clock, diverse quantità e configurazioni di memoria on-chip, diverse interfacce (tra cui la PCI).
I modelli che vedremo più da vicino sono tre e differiscono principalmente per frequenza di clock e tipo di aritmetica (fixed point oppure floating point), differenza sostanziale già sottolineata nel corso della trattazione generale:

- **C62x**: aritmetica intera, clock 150-200 MHz

netwaves.it

SISTEMI PER LA COMUNICAZIONE DATI, PROGETTAZIONE ELETTRONICA E RICONOSCIMENTO VOCALE

DEVICE NETWORKING

SENA TECHNOLOGIES

Sistemi per la conversione da Seriale a Ethernet. Versioni su modulo per applicazioni embedded, su scheda open frame ideali per gli integratori, e versioni esterne boxate implementabili in qualsiasi apparecchiatura dotata di porta RS232. Diverse caratteristiche tecniche offerte da una gamma di prodotti completa ed efficace anche dal punto di vista economico. Soluzioni a partire da €19.

BLUETOOTH

HandyWave FREE2MOVE

Convertitori da seriale a Bluetooth. Versioni su modulo per applicazioni embedded con diverse soluzioni tecniche orientate all'ottenimento di un prodotto finale ottimizzato come dimensioni e prestazioni.
Disponibilità di dongle esterni collegabili direttamente alla porta RS232 delle apparecchiature o PLC per connessioni wireless "PLUG & PLAY", ideale per il machine to machine.

RICONOSCIMENTO E SINTESI VOCALE

SENSORY

Chip e sistemi per il riconoscimento e la sintesi vocale. Disponibili starter kit e sistemi di sviluppo ottimali per lo sviluppo di applicazioni con le nuove tecnologie a riconoscimento vocale senza grossi investimenti iniziali.

INTERFACCE USB E PCI

MOSCHIP SEMICONDUCTOR

Circuiti integrati per l'implementazione di interfacce USB e PCI all'interno delle proprie schede. Disponibilità gratuita di tutto il supporto software (driver) necessario per la gestione della porta USB sotto i sistemi Windows.

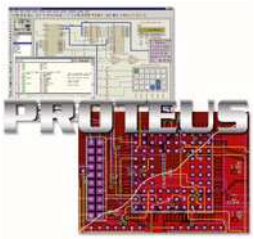
SISTEMI DI SVILUPPO PER PICmicro

CCS HI-TECH MikroElektronika

Evaluation board, programmatori, in circuit debuggers, compilatori C, Basic e Pascal. Tutti prodotti best seller sia per un impiego educational, hobbistico ma anche professionale.

CAE / CAD

labcenter Electronics



PROTEUS, pacchetto software per la stesura degli schemi, simulazione e sbroglio dei circuiti. Suite completa ed efficiente, unica nel suo genere a fornire la simulazione del circuito in modalità mixed-mode con simulazione contemporanea del microcontrollore.

Per informazioni: Tel. **+39 0266504794** / **+39 0266504755** - Fax **+39 0266508225** - Http://**www.netwaves.it** - Email: **info@netwaves.it**

- **C67x**: floating point, clock 100-166 MHz
- **C64x**: aritmetica intera, clock 300-600 MHz

Le coppie C62x, C64x e C62x, C67x sono code compatible, cioè compatibili tra loro come librerie e nell'utilizzo.
Ci interessiamo di questo specifico tipo di DSP perché rappresenta un'implementazione valida ed interessante delle architetture VLIW.
I TMS320C6000 sono caratterizzati dall'architettura *VelociTI*, un'architettura VLIW ad elevatissime prestazioni, altamente deterministica, che ha poche restrizioni su come o quando le istruzioni sono prelevate, eseguite o memorizzate. È proprio questa flessibilità architetturale che aumenta l'efficienza del relativo compilatore C ottimizzato.
I TMS320C6000 eseguono fino a 6000 milioni di istruzioni per secondo (MIPS), prediligono il compilatore C e, grazie alle loro elevate prestazioni, alla facilità d'uso e ad un costo relativamente basso, rappresentano la soluzione ideale per le seguenti applicazioni: modem via cavo, sistema telefonico multi canale, radar, modellazione atmosferica, imaging, GPS, diagnosi medica remota, audio, analisi ad elementi finiti (FEM), eccetera.

Funzionalità avanzate della tecnologia VelociTI

Le funzionalità avanzate specifiche dell'architettura VelociTI sono:
- Impacchettamento delle istruzioni, ogni istruzione ha però una dimensione estremamente ridotta.
- Tutte le istruzioni possono operare condizionatamente, cioè all'interno dell'istruzione è prevista la condizione di esecuzione, cosa che rende molto flessibile il codice.
- Istruzioni a larghezza variabile, caratteristica di flessibilità dei tipi di dati.

Descrizione Generale

Ogni chip appartenente alla famiglia di processori C6000 si basa su 32 registri general-purpose a 32 bit e 8 unità logiche:
- 4 ALU che possono lavorare sia in floating point che in fixed point.
- 2 ALU fixed point.
- 2 moltiplicatori che possono lavorare sia in floating point che in fixed point.

Dispone di un grosso banco di memoria SRAM sul chip e di un vasto set di periferiche. La memoria on-chip è suddivisa in due banchi:
- Memoria di programma.
- Memoria dati.

Diagramma Funzionale

La cosa interessante che risalta nello schema in figura 5 è la presenza di due controllori per la memoria dati e la memoria programmi ed in particolare è interessante il blocco EMIF, che rappresenta un sistema di management, in quanto gestisce sia il controller della memoria programmi, sia il controller della memoria dati, sia il DMA controller.

Architettura

Abbiamo la cache divisa in cache istruzioni e cache dati, poi altri blocchi importanti. Ma concentriamoci sulla CPU: c'è un'unità di fetch, un'unità per il dispatching delle istruzioni, quindi un blocco di decodifica delle istruzioni e infine due data path separati, a cui corrispondono due banchi di registri, ognuno dei quali va alimentare 4 unità funzionali indicate con L, S, M, D. Queste unità funzionali sono dedicate ad effettuare un determinato tipo di operazione. Quindi ci sono registri di controllo e logici, un blocco per la gestione delle interruzioni, eccetera.

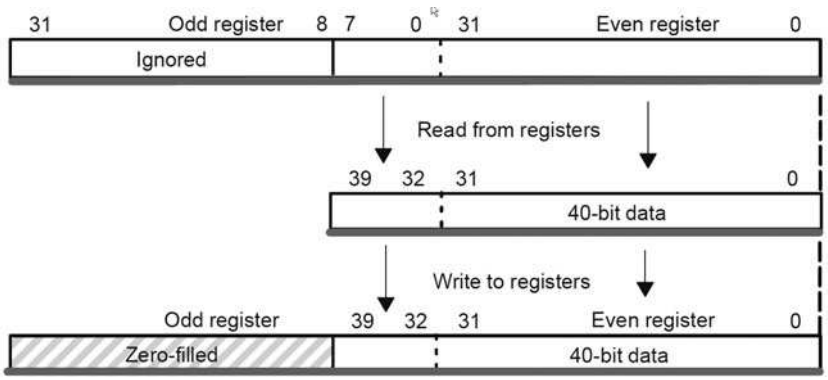


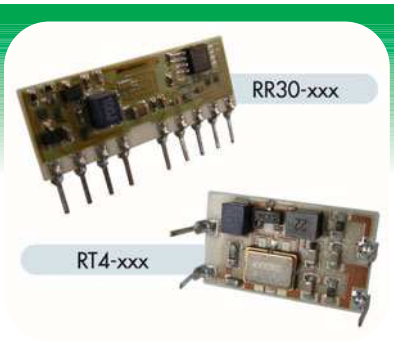
Figura 7 Memorizzazione di un dato su più di 32 bit

telecontrolli

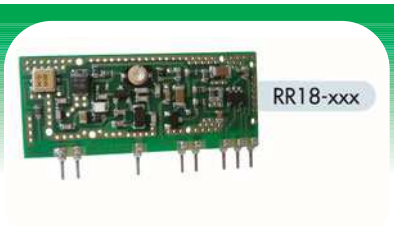
Novità

Leader nella progettazione e produzione di circuiti ibridi con tecnologia in film spesso

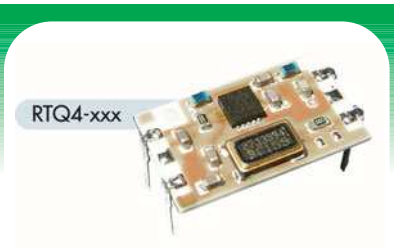
Modello	Alimentazione	Sensibilità RF/Potenza trasmissione	Frequenza (xxx)	Velocità trasmissione	Descrizione
RR30-xxx	5Vdc/2,5mA	-105dBm	300-450MHz	4,8Kbps	Coppia di moduli AM di dimensioni estremamente compatte (TX = 17,8 x 10,2mm, RX = 25,4 x 8,9mm), omologati I-ETS 300-220, idonei per applicazioni di controllo remoto
RT4-xxx	2÷14Vdc/4mA	+7dBm	303,8-433,92MHZ	4Kbps	



Modello	Alimentazione	Sensibilità RF	Frequenza (xxx)	Velocità trasmissione	Descrizione
RR18-xxx	3Vdc/70mA	-96dBm	433,92MHz	4,8Kbps	Ricevitore AM superigenerativo a basso consumo, con filtro saw in ingresso e banda stretta

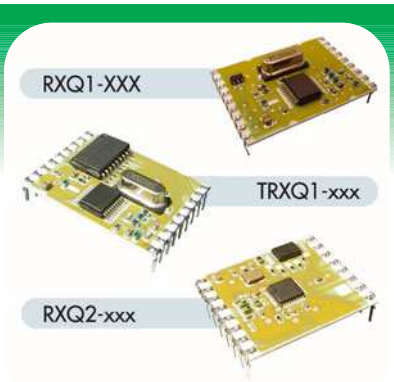


Modello	Alimentazione	Potenza trasmissione	Frequenza (xxx)	Velocità trasmissione	Descrizione
RTQ4-xxx	2÷5,5Vdc/12mA	+7dBm	433,92MHz 868,35MHz 915MHz	9,6Kbps	Trasmettitori AM con oscillatore al quarzo, dimensioni compatte (17,78 x 10,16 mm) e pin out compatibile con il modello RT4



Modello	Alimentazione	Potenza trasmissione	Frequenza (xxx)	Velocità trasmissione	Descrizione
RTFQ4-xxx	2÷5,5Vdc/12mA	+7dBm	433,92MHz 868,35MHz 915MHz	9,6Kbps	Trasmettitore FM con oscillatore al quarzo, dimensioni compatte (17,78 x 10,16 mm) e pin out compatibile con il modello RT4

Modello	Alimentazione	Sensibilità RF	Frequenza	Velocità trasmissione	Descrizione
RXQ1-XXX	2.7÷5,25Vdc/12mA	-100dBm	433,92MHz 434,33MHz	20Kbps	Transceiver a 2 canali
TRXQ1-XXX	2.7÷5,25Vdc/12mA	-100dBm	433,92MHz 434,33MHz	20Kbps	Transceiver a 2 canali con encoder/decoder
RXQ2-xxx	2÷3,6Vdc	-100dBm	433,92MHz	38,4Kbps	Transceivers multicanale per trasmissione dati veloci e sicure Completo d'interfaccia RS-232
	2÷3,6Vdc	-100dBm	868,35MHz	38,4Kbps	



Maggiori informazioni sono disponibili nel sito **www.telecontrolli.com**,
campioni disponibili per vendita on-line tramite **www.elettroshop.com** nella sezione "Moduli RF".

Nel particolare di figura 6 abbiamo i due data path, con ognuno un banco di 16 registri general purpose e 4 unità funzionali alimentate da tali registri. Le operazioni possibili sulla memoria sono di tipo load e store, abbiamo quindi due load e due store per ogni data path, cioè si può accedere alla memoria con due load o due store contemporanei, cosa che abbiamo visto indicato nelle caratteristiche generali dei DSP.
Poi abbiamo due data path incrociati tra i register files, che ci permettono di spostare i dati da una parte all'altra (1X e 2X).
Per ogni unità funzionale abbiamo due ingressi (src1 e src2) di cui il primo operando viene sempre dal banco di registri corrispondente, mentre il secondo operando a seconda che sia attivo o meno il data path incrociato può provenire dal banco di registri A o B.
Quindi abbiamo per alcune unità funzionali un ingresso "long src", che rappresenta un ingresso aggiuntivo a 8 bit per le unità che lavorano su floating point e supportano dati a 40 bit.
Ogni unità funzionale è specializzata su determinate operazioni, abbiamo per esempio:

- **L:** operazioni in virgola fissa, operazioni aritmetiche di confronto, operazioni logiche e conteggio normalizzato.
- **S:** operazioni aritmetiche a 32 bit, shift di 32 o 40 bit, operazioni logiche, generazione di costanti.
- **M:** prodotti 16 bit per 16 bit.
- **D:** operazioni su indirizzi, indirizzamento lineare e circolare, load e store con offset.

I due register files (A e B) sono costituiti ciascuno da 16 registri da 32-bit (A0-A15 per il file A e B0-B15 per il file B) per il C62x e C67x, mentre il C64x ne ha un numero doppio (A0-A31 per il file A e B0-B31 per il file B). I dati supportati sono compresi nel range 16-40 bit (come già accennato) per i fixed point e 64 bit per floating point.
Valori più grandi di 32 bit, come ad esempio 40 o 64 bit, sono immagazzinati in coppie di registri, secondo lo schema di figura 7.
I 32 bit meno significativi (LSB) del dato sono memorizzati in un registro dispari e i restanti 8 o 32 bit più significativi (MSB) nel registro immediatamente superiore (che è sempre un registro pari).

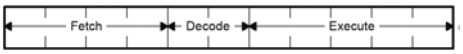


Figura 8 Fixed-Point Pipeline stages

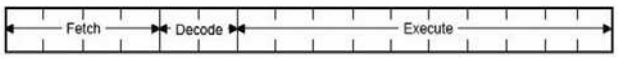


Figura 9 Floating-Point Pipeline stages

Pipeline

Prima di approfondire questo aspetto ricordiamo che il pipelining è una tecnica utilizzata nei microprocessori, che consente di lavorare più istruzioni contemporaneamente. In particolare il microprocessore inizia ad lavorare (secondo il proprio ciclo fondamentale) una seconda istruzione prima ancora che la precedente sia stata completata. Le istruzioni si trovano simultaneamente nella Pipeline, ognuna ad un diverso stato di elaborazione. I risultati finali di ogni istruzione emergono alla fine della Pipeline in rapida successione.
Questa architettura, sebbene caratterizzata da più unità di elaborazione, è comunque organizzata secondo pipeline. La pipeline è progettata per fornire flessibilità al fine di semplificare la programmazione ed aumentare le prestazioni.
Due fattori concorrono all'ottenimento di questo obiettivo:

- Il controllo della pipeline è semplificato mediante l'eliminazione di possibili stalli.
- L'aumento del pipelining elimina i colli di bottiglia delle architetture tradizionali nella fase di fetch, accesso ai dati e operazioni di moltiplicazione. Ciò consente un'istruzione eseguita per ogni ciclo.

Le caratteristiche generali delle pipeline sono:

- Le pipeline possono servire otto istruzioni parallelamente ad ogni ciclo di clock.
- Le istruzioni parallele procedono simultaneamente attraverso ogni fase della pipeline.
- Le istruzioni seriali procedono attraverso la pipeline con un ritardo fisso tra le istruzioni.
- Gli indirizzi di Load e Store sono visibili alla CPU durante la stessa fase della pipeline, eliminando i conflitti read-after-write.

- Tutte le istruzioni richiedono lo stesso numero di fasi per il fetch e il decode, ma per l'execute il numero di fasi cambia in base al tipo di istruzione.

Le fasi della pipe sono raggruppate in tre stadi:
- **Fetch**: ha quattro fasi per tutte le istruzioni.
- **Decode**: ha due fasi per tutte le istruzioni.
- **Execute**: ha un numero variabile di fasi, dipendente dal tipo di istruzione (Es. Fixed-Point e Floating-Point).

Vediamo le singole fasi dei diversi stage:
- **Fetch Stage**, composto dalle seguenti fasi:
  - **PG** (Program address Generate)
  - **PS** (Program address Send)
  - **PW** (Program access ready Wait)
  - **PR** (Program fetch packet Receive)

  In pratica queste fasi servono per comunicare con la memoria ed al termine dell'ultima fase il pacchetto di istruzioni è presente all'interno della pipe.
  In ogni ciclo un nuovo pacchetto di otto istruzioni entra nella fase PG dello stage di Fetch. Contemporaneamente il pacchetto che si trova nella fase PR viene inviato allo stage Decode.
- **Decode Stage**, composto dalle seguenti fasi:
  - **DP** (Instruction Dispatch): fase in cui, dai 256 bit di cui era composto il pacchetto di fetch, si ricavano i pacchetti di esecuzione con le istruzioni in parallelo.
  - **DC** (Instruction decode): ad ogni istruzione vengono associate le sorgenti e le destinazioni (registri o indirizzi di memoria).

  Nella fase DP i pacchetti provenienti dal Fetch stage sono convertiti in pacchetti di esecuzio-

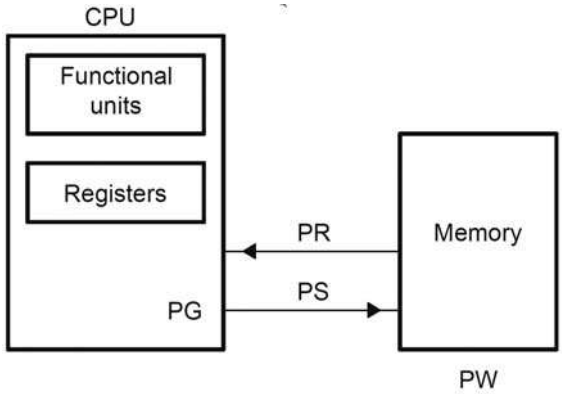


Figura 10 Fetch Stage

circuiti stampati in 24 ore
garantiamo il tempo di consegna: 24 ore o i circuiti sono gratis
Potrete scegliere tra singola e doppia faccia con foro metallizzato. Con solder e serigrafie per uno stampato di alta qualità o solo piste stagnate per un prototipo a basso costo.
Prezzi a partire da* € 14,38 (doppia faccia foro metallizzato 7,50x7,50 cm) e da € 9,13 (singola faccia 7,50x7,50 cm) per FR4 1,6 mm con rame 35 µm, tutti comprensivi di attrezzatura.
Nessuna limitazione sul numero dei fori, sul numero degli utensili (diametri) e sul tipo di scontornatura (anche tondeggiante).
Distanza minima tra le piste e pista minima 8 mils (0,20 mm).
PREVENTIVO ANONIMO, GRATUITO ED IMMEDIATO con il nostro calcolatore online.
+ QUALITÀ - TEMPO
visita il nostro sito per il dettaglio delle note tecniche
www.mdsrl.it
millennium md dataware
millennium dataware srl
parco scientifico e tecnologico
15050 rivalta scrivia - tortona (al)
tel. 0131 860.254 fax 0131 860157
www.mdsrl.it info@mdsrl.it
* i prezzi si intendono iva esclusa e calcolati sul singolo pezzo - ordine minimo 2 pezzi

ne. Ogni pacchetto di esecuzione contiene da un minimo di una ad un massimo di otto istruzioni da eseguire in parallelo, si parla di minimo e massimo in quanto non è detto che tutte le istruzioni decodificate possano essere inviate in parallelo nelle unità funzionali senza generare conflitti, per esempio in figura 12 abbiamo 5 istruzioni di cui una NOP, che di solito viene utilizzata quando bisogna aspettare o per evitare che possano esserci conflitti. Le istruzioni che compongono il pacchetto di esecuzione sono associate alle opportune unità funzionali. Nella fase DC, in cui avviene la vera e propria decodifica delle istruzioni, vengono individuati i registri sorgente e destinazione e i segnali di controllo del Data path.

- **Execute Stage**

Lo stage Execute, verrà descritto nel caso di operazioni floating point (solo per C67x), è suddiviso in dieci fasi (E1-E10).
Differenti tipi di istruzioni richiedono un numero differente di queste fasi per completare la loro esecuzione. Tale stage segue lo schema in figura 14.

Abbiamo le 8 unità funzionali e i due banchi di registri generali. I dati vengono caricati da i registri oppure direttamente dalla memoria attraverso load e store.
La pipeline a regime assume l'aspetto come in figura 15.

Per ogni Fetch Packet si ha un unico Execute Packet. A regime tutto dovrebbe essere regolato da una corrispondenza 1 a 1, ma nella realtà dei fatti non è mai così per i possibili stalli che si possono verificare prima delle unità funzionali.
Ogni istruzione all'interno del Fetch Packet contiene un bit (p-bit), inserito dal compilatore in fase di compilazione.
Questo p-bit è una traccia lasciata dal compilatore sulla possibilità di poter eseguire o meno in parallelo l'istruzione i considerata e la successiva i+1.
I p-bit vengono testati a partire da sinistra. Se il p-bit dell'istruzione ***i*** è alto l'istruzione ***i+1*** può essere eseguita in parallelo all'istruzione ***i***; altrimenti l'istruzione ***i+1*** deve essere eseguita nel ciclo successivo all'istruzione ***i***.

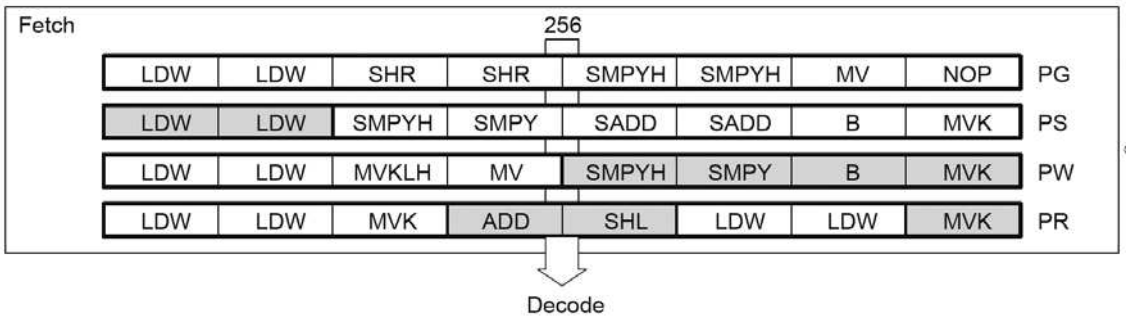


Figura 11 *Le istruzioni che vengono caricate con parallelismo di 256 bit nella fase di PG*

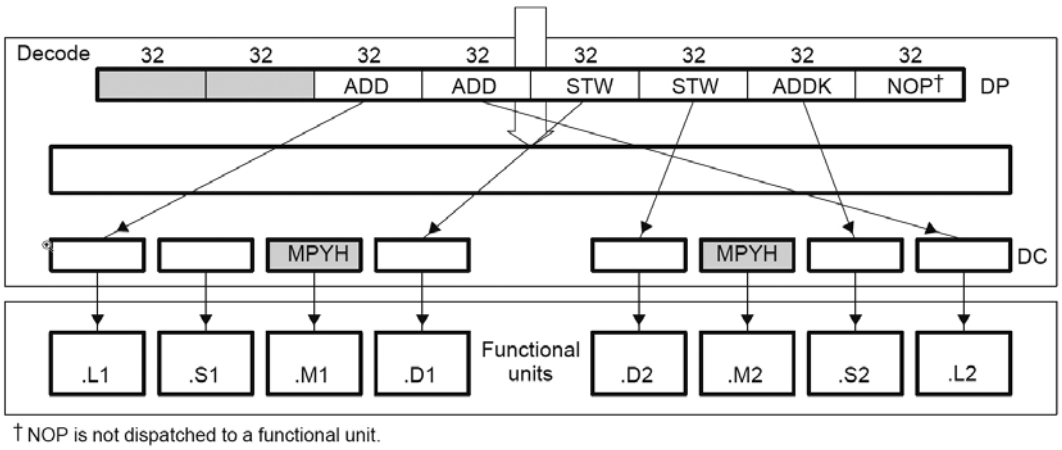


Figura 12 *Definizione degli operandi e dell'unità funzionale relativa all'istruzione*

E1	E2	E3	E4	E5	E6	E7	E8	E9	E10

Figura 13 10 Fasi dell'Execute Stage

Insomma una cosa in qualche modo assimilabile al bit a 0 del collision vector nelle architetture superscalari.

Istruzioni particolari – Nop multiciclo

Concludiamo questo breve cenno alla famiglia dei DSP della Texas Instruments TMS320 con un'istruzione particolare, la nop multiciclo. Ipotizziamo di avere le 4 istruzioni in figura 17. Abbiamo un'operazione di load, una di addizione, una di moltiplicazione e una nop.
Il load, per esempio, è visibile dopo 5 cicli, l'operazione di add è visibile dopo 1 ciclo, mentre il risultato della moltiplicazione è visibile dopo 2 cicli, tutto questo perché le diverse istruzioni hanno fasi di esecuzione diverse. La nop multiciclo elimina i conflitti di tipo read after write.
Includendo dopo n istruzioni la nop, tale operazione fa in modo che i risultati di tutte le operazioni siano disponibili con un ritardo pari al massimo ritardo relativo alle diverse istruzioni precedenti.
Quindi in questo caso tutti i risultati sono visibili dopo 5 cicli.
Se c'è da fare un add con i dati forniti dall'operazione di load, tale operazione potrà essere effettuata solo quando tali dati saranno disponibili a seguito del load, quindi senza alcun problema di tipo read after write. In tal caso la pipe non va in stallo, ma viene ritardata.

BIBLIOGRAFIA

G. Hennessy e D.A. Patterson, Computer Architectures: a Quantitative Approach. Ed. Morgan – Kaufmann.
H. S. Stone, High Performance Computer Architecture, Ed. Addison-Wesley.
Andrew S. Tanenbaum, Structured Comter Organization, Prentice-Hall.
Diagrammi a blocchi e informazioni tratte da datasheet circa la famiglia Texas Instruments DSP TMS320C6000 CPU and Instruction Set Reference Guide, tratto da:
http://dspvillage.ti.com

Abbonati oggi!

www.farelettronica.com/abbonamento

fare elettronica

CULTURA ELETTRONICA APPLICATA

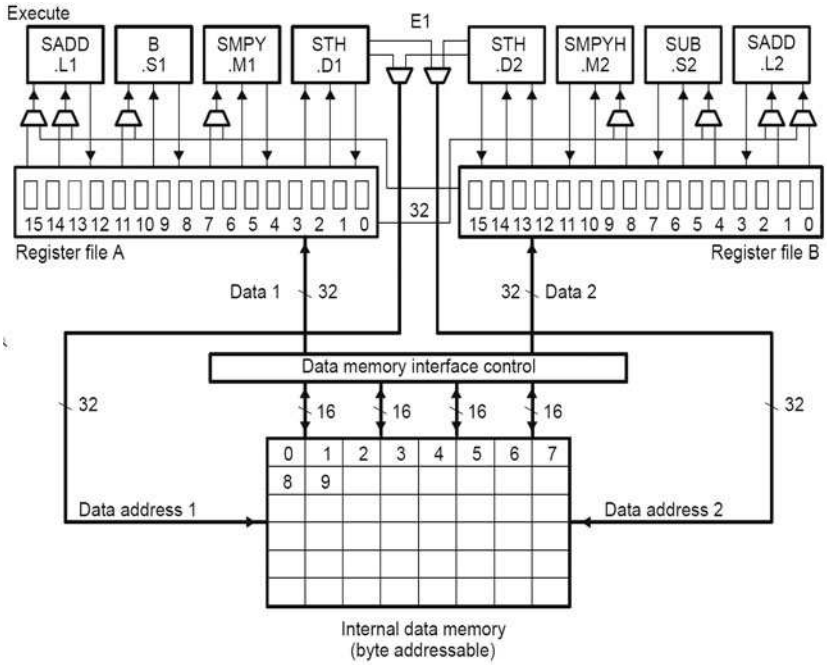


Figura 14 Execute stage

CONCLUSIONI

Abbiamo analizzato alcune delle caratteristiche principali che contraddistinguono questa tipologia di processori.
L'argomento è molto vasto e di conseguenza complesso.
In questa semplice introduzione si è cercato di dare solamente un'idea, fornendo un esempio per certi versi fin troppo approfondito in modo da suscitare l'interesse nei riguardi di quanto si trova dietro i sofisticati dispositivi che utilizziamo tutti i giorni.
Arrivederci alla prossima occasione di approfondimento.

packet	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
n	PG	PS	PW	PR	DP	DC	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	
n+1		PG	PS	PW	PR	DP	DC	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10
n+2			PG	PS	PW	PR	DP	DC	E1	E2	E3	E4	E5	E6	E7	E8	E9
n+3				PG	PS	PW	PR	DP	DC	E1	E2	E3	E4	E5	E6	E7	E8
n+4					PG	PS	PW	PR	DP	DC	E1	E2	E3	E4	E5	E6	E7
n+5						PG	PS	PW	PR	DP	DC	E1	E2	E3	E4	E5	E6
n+6							PG	PS	PW	PR	DP	DC	E1	E2	E3	E4	E5
n+7								PG	PS	PW	PR	DP	DC	E1	E2	E3	E4
n+8									PG	PS	PW	PR	DP	DC	E1	E2	E3
n+9										PG	PS	PW	PR	DP	DC	E1	E2
n+10											PG	PS	PW	PR	DP	DC	E1

Figura 15 Pipeline a regime

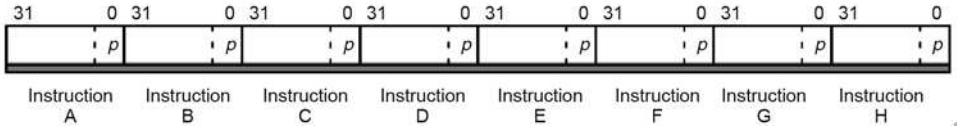


Figura 16 I p-bit

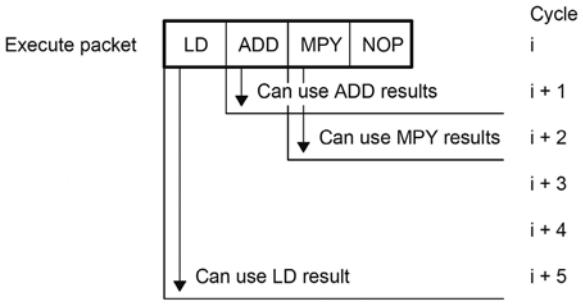


Figura 17 Nop multiciclo

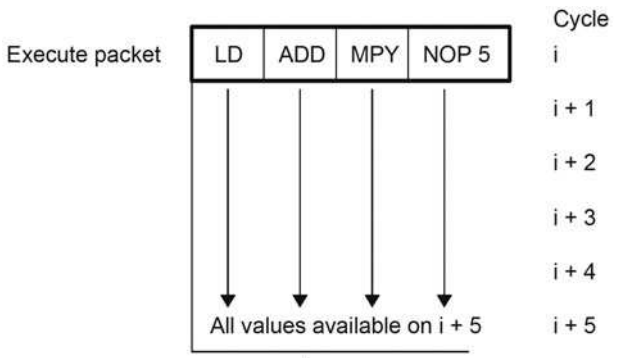


Figura 18 Effetti Nop multiciclo

Sistemi professionali GPS/GSM

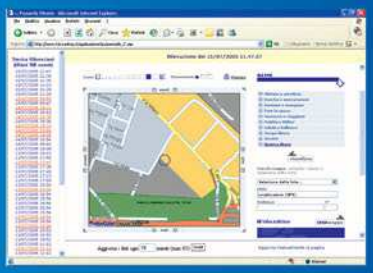
Produciamo e distribuiamo sistemi di controllo e sorveglianza remoti basati su reti GSM e GPS. Oltre ai prodotti standard illustrati in questa pagina, siamo in grado di progettare e produrre su specifiche del Cliente qualsiasi dispositivo che utilizzi queste tecnologie. Tutti i nostri prodotti rispondono alle normative CE e RTTE.

Localizzatore GPS/GSM portatile

FT596K - Euro 395,00

Unità di localizzazione remota GPS/GSM di dimensioni particolarmente contenute ottenute grazie all'impiego di un modulo Wavecom Q2501 che integra sia la sezione GPS che quella GSM. L'apparecchio viene fornito premontato e comprende il localizzatore vero e proprio, l'antenna GPS, quella GSM ed i cavi adattatori d'antenna. La tensione di alimentazione nominale è di 3,6V, tuttavia è disponibile separatamente l'alimentatore switching in grado di funzionare con una tensione di ingresso compresa tra 5 e 30V (FT601M - Euro 25,00) che ne consente l'impiego anche in auto. I dati vengono inviati al cellulare dell'utente tramite SMS sotto forma di coordinate (latitudine+longitudine) o mediante posta elettronica (sempre sfruttando gli SMS). In quest'ultimo caso è possibile, con delle semplici applicazioni web personalizzate, sfruttare i siti Internet con cartografia per visualizzare in maniera gratuita e con una semplice connessione Internet (da qualsiasi parte del mondo) la posizione del target e lo spostamento dello stesso all'interno di una mappa. A tale scopo, unitamente al localizzatore, vengono forniti i listati esemplificativi di alcune pagine web da utilizzare per creare una connessione Internet personalizzata. Il dispositivo viene fornito premontato.

FT596K (premontato) - Euro 395,00
FT601M (montato) - Euro 25,00



Localizzatore GPS/GSM con ambientale

Apparato di controllo a distanza GPS/GSM in grado di stabilire la posizione di un veicolo e di ascoltare quanto viene detto all'interno dello stesso. Il sistema è composto da un'unità remota (montata sulla vettura) e da una stazione base che utilizza un PC, un'apposito software di connessione, un software cartografico con le mappe dettagliate di tutta Italia ed un modem GSM per il collegamento. Per l'ascolto ambientale è sufficiente l'impiego di un telefono fisso o di un cellulare.

Unità base

Il REM2004 comprende tutti gli elementi hardware e software necessari per realizzare una stazione base con la quale visualizzare in tempo reale la posizione di un'unità remota GSM/GPS, scaricare i dati relativi al percorso, programmare tutte le funzioni, visualizzare i dati storici, eccetera. L'unico elemento non compreso è il PC. Il software di gestione è compatibile con l'unità remota con memoria FT521K. Per la connessione all'unità remota questo sistema utilizza un modem GSM che deve essere reso attivo con l'inserimento di una SIM card valida. La SIM card non è compresa. Il set REM2004 è composto dai seguenti elementi:

- Modem GSM bibanda GM29;
- Antenna a stilo GSM bibanda con cavo di connessione;
- Alimentatore da rete per modem GM29;
- Cavo seriale DB9/DB9 per collegamento al PC;
- Software di connessione e gestione REM2004 (SFW521);
- Software di gestoine cartografica Fugawi 3.0 con chiave hardware (USB);
- CD con mappe stradali di Italia, Svizzera e Austria EUSTR2).

Disponibili mappe dettagliate di tutta Europa.

REM2004 - Euro 560,00

FT521 - Euro 480,00

Unità remota

Compatta unità remota di localizzazione e ascolto ambientale che utilizza le reti GPS e GSM per rilevare la posizione del veicolo e trasmettere i dati alla stazione di controllo. Il circuito dispone inoltre di un sistema di ascolto ambientale. L'unità remota comprende anche il ricevitore GPS con antenna integrata, l'antenna GSM ed il microfono preamplificato. Il dispositivo viene fornito montato e collaudato.

Caratteristiche elettriche generali

Alimentazione 12 VDC; Assorbimento a riposo: 110 mA (GPS attivo); Assorbimento in collegamento: 380/480 mA; Memoria dati: 8.192 punti; Sensibilità microfonica max -70 dB; Dimensioni: 35 x 70 x 125 mm (esclusa antenna GPS); Sensore di movimento al gas di mercurio.

Funzionalità

Completamente teleconfigurabile; Password di accesso; Funzionamento in real time; Memorizzazione dati su remoto (8.192 punti); Tempo di polling regolabile; Sensore di movimento programmabile; Attivazione GPS programmabile; SMS di allarme gestito da sensore di movimento; Verifica tensione di batteria con gestione SMS di allarme; Ascolto ambientale configurabile da remoto.

Tutti i prezzi si intendono IVA inclusa.

Telecontrollo GSM bidirezionale

FUTURA ELETTRONICA

Via Adige, 11 -21013 Gallarate (VA)
Tel. 0331/799775 - Fax. 0331/778112
www.futuranet.it

Maggiori informazioni su questi prodotti e su tutti le altre apparecchiature distribuite sono disponibili sul sito *www.futuranet.it* tramite il quale è anche possibile effettuare acquisti on-line.

Unità di controllo remoto GSM con due ingressi fotoaccoppiati e due uscite a relè. Utilizzabile sia per attivare a distanza qualsiasi apparecchiatura che per ricevere messaggi di allarme. In modalità apricancello è in grado di memorizzare fino ad un massimo di 100 utenti. Ideale per realizzare impianti antifurto per abitazioni e attività commerciali, car alarm, controlli di riscaldamento/condizionamento, attivazioni di pompe e sistemi di irrigazione, apertura cancelli, controllo varchi, circuiti di reset, ecc. Fornito montato e collaudato.

Caratteristiche tecniche:

Frequenza di lavoro: GSM bibanda 900/1.800MHz; Funzione apricancello a costo zero; Ingressi optoisolati: 2; Uscite a relè (bistabile o astabile): 2; Numeri abbinabili per allarme: 5; Numeri abbinabili per apricancello: 100; Carico applicabile alle uscite: 250V, 5A; Alimentazione: 5÷32V; Assorbimento massimo: 550mA.

STD32 - Euro 228,00

Settima parte
n° 245 - Novembre 2005
Pilotiamo i motori passo-passo

Ottava parte
n° 246 - Dicembre 2005
Ambiente di sviluppo

Nona parte
n° 247 - Gennaio 2006
Come si memorizzano i dati e le informazioni

Mikrobasic per PICmicro

In questa puntata prendiamo confidenza con il potente ambiente IDE del compilatore Mikrobasic, per ottenere i massimi risultati nella programmazione assistita.

Dopo aver analizzato numerosi aspetti della programmazione del linguaggio Basic applicato ai microcontrollori (nella fattispecie al PIC16F84 della Microchip), con questo nuovo articolo studiamo le funzionalità che fanno parte dell'interfaccia utente del compilatore MikroBasic, meglio conosciute con il nome di **IDE**.

PERCHÉ ADESSO?

Abbiamo preparato questo articolo accontentando le numerose richieste giunte dai lettori. Perché non lo abbiamo pubblicato prima?
Innanzi tutto perché, con la lettura del presente, si ha possibilità di prendere un po' "di fiato" dopo tante nozioni di teoria e pratica, con la possibilità di studiare ed approfondire ancora meglio quanto già trattato nelle precedenti puntate.
In secondo luogo, se questo articolo fosse stato presentato all'inizio del corso, la maggior parte di lettori, in special modo quelli alle prime armi, non avrebbero potuto apprezzarne i contenuti perché ancora troppo "impreparati" per acquisire certe informazioni che invece avrebbero potuto costituire, con tutta probabilità, motivo di confusione o poca chiarezza. Pertanto la collocazione dell'articolo in questo punto del corso è giusta al fine di approfondire quello che, a detta di molti, è uno dei migliori compilatori Basic per PICmicro sul mercato.

IDE DEL MIKROBASIC

Il Mikrobasic costituisce un completo ed efficiente ambiente **IDE** (**I**ntegrated **D**evelopment **E**nvironment), infatti in un unico pacchetto software sono compresi: l'editor del sorgente, il compilatore Basic, il debugger e l'interfaccia di controllo del programmatore. Sono anche presenti numerose utility che aiutano il programmatore nella realizzazione e nella messa a punto di algoritmi critici e di programmi complicati.
Elenchiamo adesso solo alcune potenzialità del pacchetto software in questione:

- È un ambiente di sviluppo davvero completo.
- Dispone di un utilissimo convertitore in notazione HEX, BIN e DEC.
- Dispone di un Code Explorer avanzato.
- Presenta un ottimo assistente di codice.
- Permette la colorazione del codice sorgente contestualmente alla sua sintassi.
- Possiede un esauriente Help on-line.
- Genera automaticamente il file HEX.
- Visualizza il codice Assembler del programma.
- Mostra le statistiche (grafiche) sull'uso delle risorse.
- Presenta molte funzioni di utilità.
- ... e molto ancora.

Dopo questa introduzione possiamo asserire che il MikroBasic dispone di caratteristiche avanzate che ne giustificano l'acquisto, visto anche che il prezzo non è elevato ed, in definitiva, adatto a tutte le tasche.
Il presente articolo si riferisce alla *release 2.0.0* che si è dimostrata all'altezza della situazione e che sicuramente sarà soggetta a migliorie ed aggiornamenti continui, come del resto accade a tutti i compilatori prodotti dalla Mikroelektronika. Ricordiamo il MikroBasic ha due *fratelli*: il MikroC (compilatore C) e MikroPascal (compilatore Pascal) oggetto di un corso che sarà presentato nel corso del 2006.

Ambiente di sviluppo

di Giovanni Di Maria
(g.dimaria@farelettronica.com)

L'INSTALLAZIONE

Certamente semplicissima da eseguire, l'installazione del pacchetto non comporta alcun problema di sorta, e può essere eseguita anche su personal computer datati o, comunque, non eccezionalmente potenti.
Il primo passo da compiere è quello di scaricare il file di installazione dal sito di Fare Elettronica (Shop/Software/MikroBasic). Si tratta della versione demo, completamente funzionante ma con una sola limitazione: non compila programmi con dimensioni superiori a 2K (program words). Si esegua pertanto, tramite doppio click, l'installazione della procedura.
Dopo il messaggio di benvenuto, il sistema avvisa dell'inizio dell'installazione e consiglia di chiudere tutte le altre procedure eventualmente aperte. Chiaramente bisogna confermare cliccando il pulsante **Next**.
La seconda schermata invita l'utente alla lettura ed approvazione della licenza d'uso, per proseguire occorre accettare i termini della licenza, abilitando la prima opzione (I' accepts…"), e confermare la pagina cliccando nuovamente sul pulsante **Next**.
Si accede pertanto alla terza maschera, che chiede quali componenti del pacchetto si intendono installare. Il compilatore ed i file di help non possono essere naturalmente omessi dal processo, mentre si è liberi di copiare o meno i file di esempio. Consigliamo in ogni caso di installarli, in quanto una loro consultazione potrebbe essere d'aiuto durante la stesura dei propri programmi. Selezionate pertanto tutte le opzioni e cliccate sul pulsante **Next**.
Nella quarta maschera l'utente è invitato a scegliere la cartella nella quale copiare ed installare il programma. Vi consigliamo, a meno che non abbiate esigenze particolari, di accettare la cartella proposta cliccando il pulsante **Install**.
Inizia adesso la vera e propria installazione che consiste nel processo di scompattazione e di copia dei file di sistema e degli eseguibili nella cartella di destinazione. La procedura è monitorata attraverso una *progress bar* di colore blu, che avanza verso destra, mostrando la percentuale di lavoro compiuto. Finalmente l'installazione ha termine e viene mostrata l'ultima finestra di conferma. Occorre solamente cliccare sul pulsante **Finish**. Il programma risulta adesso installato sul nostro PC ed a conferma di ciò si può osservare la relativa icona sul desktop (uguale a quella mostrata a fianco).

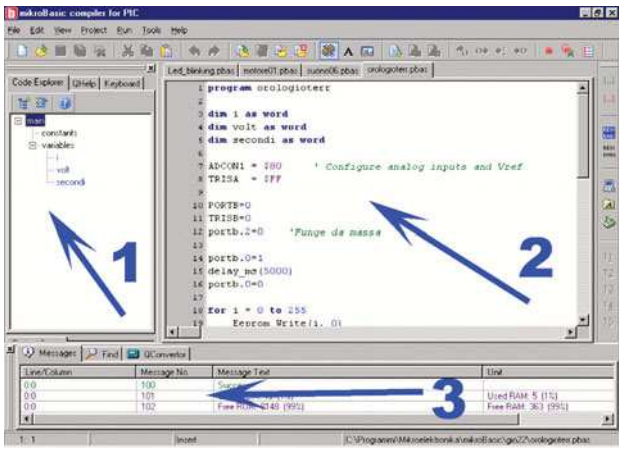


Figura 1 L'ambiente di lavoro IDE

L'ESECUZIONE

Con il doppio click sull'icona presente sul desktop, ha inizio la sessione di lavoro, preceduta dalla breve e momentanea comparsa di uno *splash screen*, che ci informa sulla versione del programma, subito dopo appare l'ambiente di lavoro, suddiviso in tante zone, ognuna delle quali svolge una precisa funzione (figura 1). Esaminiamo in dettaglio le principali:

1. La prima zona posizionata a sinistra dello schermo e denominata *Code Explorer* (esploratore), ha l'importante compito di mostrare, per il programma corrente caricato in Ram, molte informazioni relative al

codice Basic, come i nomi delle funzioni e delle procedure create ma soprattutto i nomi delle costanti e delle variabili. Si ha in questo modo la possibilità di tenere traccia di tutti i dati utilizzati nei nostri programmi. Il sottoscritto, usando altri compilatori, teneva traccia di queste informazioni semplicemente utilizzando carta e penna. Parlando sempre della prima zona, esiste la possibilità di selezionare altre due schede. La scheda **QHELP**, che mostra l'elenco alfabetico di tutte le funzioni *built-in* (funzioni di libreria e procedure), consultabile attraverso un semplice doppio click sulla voce che interessa. La scheda **Keyboard** invece fornisce un elenco completo di tutte le combinazioni dei tasti utilizzabili per accelerare una specifica operazione del compilatore (shortcuts).

2. La seconda zona, posizionata al centro dello schermo, contiene il codice Basic digitato dall'utente. Si tratta di un semplice ma potente editor, che mette a disposizione tanti utili strumenti di correzione e controllo.
3. Infine la terza zona, posizionata in basso, costituisce infine un'utile finestra dei messaggi (anche di errore) che il compilatore fornisce durante la compilazione e il linking del programma sorgente. Come si può notare, in questa sede esistono altre due schede supplementari: la scheda **Find** (trova) e la scheda **Qconvertor**, la quale ultima costituisce un potente convertitore tra i sistemi di numerazione decimale, esadecimale e binario.

IL CODE EDITOR

Come detto precedentemente, la zona numero 2 è rappresentata dal *Code Editor*, ovvero da un editor di testo dove l'utente scrive il proprio listato Basic. Vediamone le potenzialità.

Suggerimento dei parametri delle funzioni

Questa possibilità rappresenta una delle più interessanti potenzialità dell'ambiente IDE del Mikrobasic. Consiste in pratica in una sorta di suggerimento sull'utilizzo delle funzioni in Basic. Durante la digitazione di una qualsiasi funzione (numerica o non), all'apertura della parentesi tonda l'editor, mediante una bandierina gialla, suggerisce quanti e quali parametri dovranno essere immessi all'interno della funzione.

Code Assistant

Durante la digitazione di un comando, scrivendo i primi due o tre caratteri di questo e premendo contemporaneamente i tasti <CTRL> <SPAZIO>, la comparsa di una finestra agevola l'inserimento dello stesso. Per esempio, scrivendo le lettere "tr" e premendo i suddetti due tasti, comparirà una lista di possibili comandi, tutti inizianti con queste due lettere. Occorre infine posizionarsi su quello desiderato e premere Invio.

STATISTICHE

Il MikroBasic è uno dei pochi compilatori che posseggono una visualizzazione grafica delle risorse hardware utilizzate dal programma prodotto. Dopo aver compilato correttamente il codice sorgente, attraverso la scelta del menù **Project** e la voce **Build**, si può accedere al menù **View** e la voce **View Statistics**. Una finestra composta da sei schede mostrerà, molto dettagliatamente, le risorse utilizzate.
Rispettivamente le informazioni fornite (in veste grafica e testuale) riguardano:

- Utilizzo della Ram e della Rom (del PIC naturalmente) sotto forma di istogramma, di percentuale e di celle utilizzate.
- Disposizione e occupazione, sempre in veste di istogramma, delle procedure create e utilizzate nel codice sorgente.
- Occupazione "fisica" delle procedure nella memoria del PIC.
- Sommario delle funzioni utilizzate in formato gerarchico (tree).

Come si può notare si tratta di strumenti utilissimi che danno in tempo reale la completa "mappa" del lavoro prodotto (figura 2).

GLI STRUMENTI INTEGRATI

Il Mikrobasic dispone di alcune utility molto interessanti utilizzabili soprattutto per scopi di controllo e documentazione, vediamoli in dettaglio.

Usart Terminal

Si tratta di un terminale software di comunica-

Fare Elettronica sta preparando per te un nuovo bellissimo Numero Speciale

lo troverai in tutte le edicole a Gennaio*

* Se sei abbonato a Fare Elettronica richiedi la tua copia del Numero Speciale "FIRMWARE", ti sarà recapitata comodamente a casa a soli €3 anzichè €6.

Telefona subito allo 02-66504794 oppure collegati a www.farelettronica.com/firmware

zione che prevede il collegamento alla porta seriale (RS-232) del PC. Va eseguito cliccando sull'apposito menù **Tools** scegliendo la voce **Usart Terminal** (o semplicemente premendo i tasti CTRL-T). Sostituisce in pratica il programma *Hyper Terminal* di Windows, fornendo però un'interfaccia più gradevole e più intuitiva. Occorre ricordare di fornire correttamente i parametri operativi della comunicazione, quali: il numero della porta COM, la velocità di trasmissione, il numero di bit di stop, il controllo della parità e la lunghezza del pacchetto trasmesso. Può essere utilmente utilizzato per testare i modem seriali attraverso una serie di comandi **AT** (figura 3).

Ascii Chart

Selezionando dal menù **Tools** la voce **Ascii Chart** si accede all'elenco (molto importante) dei caratteri ASCII presenti nel sistema operativo ed utilizzati ormai dalla maggior parte dei compilatori e dispositivi elettronici (vedi ad esempio i display LCD intelligenti). La tabella (che visualizza tutti i 256 caratteri appartenenti al codice internazionale) contiene quattro colonne: la prima visualizza il nome fonetico del carattere (o la sua rappresentazione grafica), la seconda mostra il relativo carattere Ascii in notazione decimale, la terza lo stesso codice in notazione esadecimale e la quarta infine in notazione binaria. Molto utile quando si ha a che fare con conversioni e visualizzazioni di ogni genere (figura 4).

7 Segment Display Decoder

Si tratta di un utile tool che ha lo scopo di "comporre" la disposizione di un numero in un ipotetico display a Led a 7 segmenti, con la possibilità di ottenere la corretta rappresentazione e codificazione binaria e decimale, da dare "in pasto" al microcontrollore per la relativa visualizzazione. I codici possono essere utilizzati tanto per i display ad anodo comune quanto per quelli a catodo comune, con naturale differenziazione. Per accedere a questo tool occorre selezionare il menù Tools e la voce **Seven Segment Convertor**. La finestra che appare mostra un grosso display a 7 segmenti (inizialmente spenti). Occorre cliccare sul segmento desiderato per "illuminarlo" e per ottenere così la corretta informazione.
Si tenga presente che tale codifica è basata su un utilizzo di un bus di collegamento LED/PIC

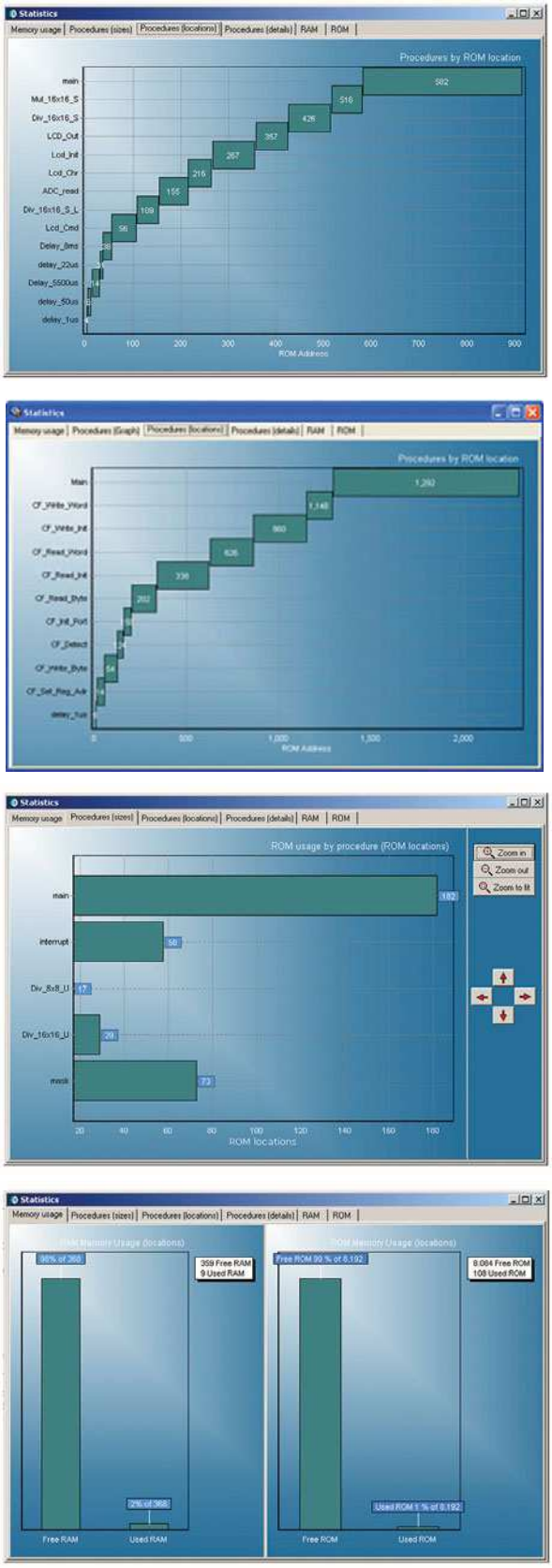


Figura 2 *Le finestre di statistica del compilatore*

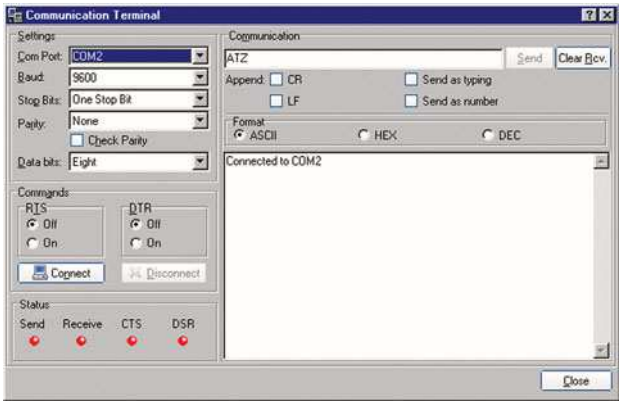
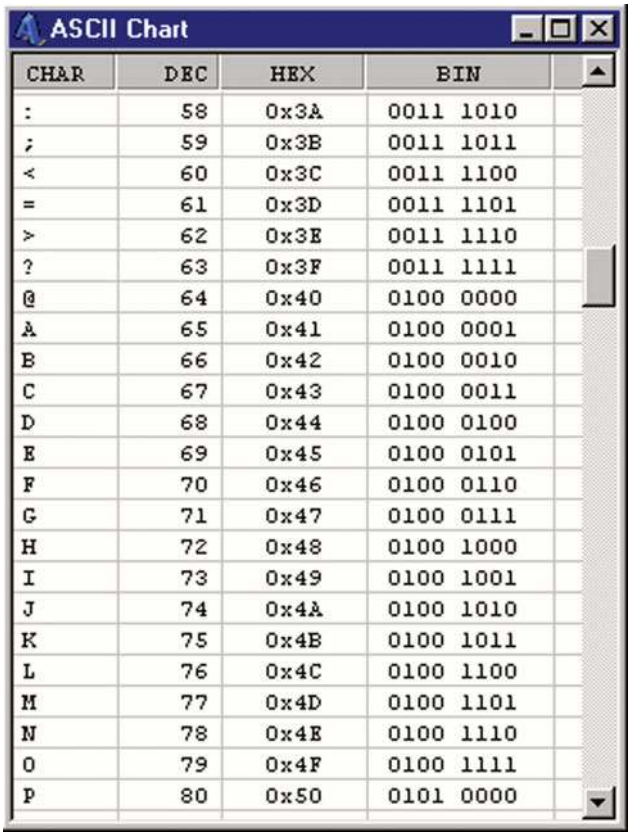


Figura 3 La finestra del terminale seriale



ASCII Chart

CHAR	DEC	HEX	BIN
:	58	0x3A	0011 1010
;	59	0x3B	0011 1011
<	60	0x3C	0011 1100
=	61	0x3D	0011 1101
>	62	0x3E	0011 1110
?	63	0x3F	0011 1111
@	64	0x40	0100 0000
A	65	0x41	0100 0001
B	66	0x42	0100 0010
C	67	0x43	0100 0011
D	68	0x44	0100 0100
E	69	0x45	0100 0101
F	70	0x46	0100 0110
G	71	0x47	0100 0111
H	72	0x48	0100 1000
I	73	0x49	0100 1001
J	74	0x4A	0100 1010
K	75	0x4B	0100 1011
L	76	0x4C	0100 1100
M	77	0x4D	0100 1101
N	78	0x4E	0100 1110
O	79	0x4F	0100 1111
P	80	0x50	0101 0000

Figura 4 L'elenco dei caratteri Ascii

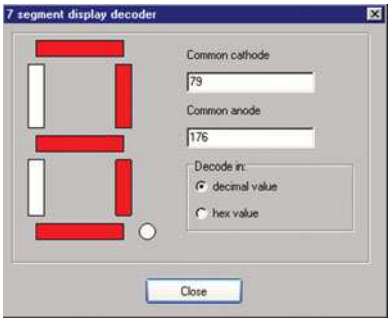


Figura 5 L'emulatore di un display a 7 segmenti con codifica

ad 8 bit, pertanto il display dovrà essere collegato ad una porta dotata di 8 pin (es. PORTB). Potrete trovare degli esempi pratici nel numero 240 di Fare Elettronica (figura 5).

MikroBootloader

I microcontrollori appartenenti alla famiglia dei PIC16F87X hanno la facoltà di modificare non solo i dati presenti nella RAM (come per il PIC16F84) ma anche quelli presenti nella FLASH. In altre parole, una routine può scrivere nella memoria programma, modificando quindi "se stessa". Questa possibilità prevede l'utilizzo di un piccolo *bootloader*, che riceve e scrive il nuovo firmware.

PicFlash Programmer

È sicuramente lo strumento più utilizzato della famiglia delle utility del compilatore. Si tratta di un programmatore software che svolge l'importante funzione del caricamento di un file .HEX, prodotto dalla compilazione, e del conseguente scaricamento nella memoria del PIC. Si accede ad esso attraverso il menù **Tools** e la voce **PicFlash Programmer** o, più semplicemente, premendo il tasto **F11** della tastiera (figura 6).

In linea generale il suo funzionamento è molto semplice. Occorre solamente caricare il file prodotto .HEX attraverso il pulsante **Load HEX**, quindi *riversarlo* nella memoria del microcontrollore premendo il pulsante **Write**. Inutile dire che bisogna collegare alla porta USB del proprio PC il relativo programmatore. In una prossima puntata tratteremo in dettaglio proprio questo aspetto.

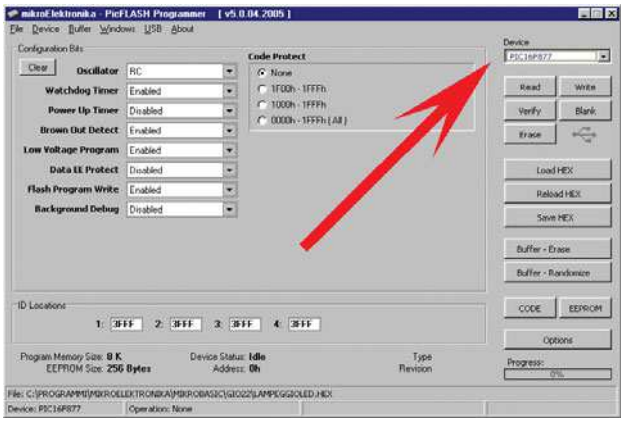


Figura 6 La finestra del programmatore PIC16F877 selezionato)

Hid Terminal
È un terminale che permette la comunicazione con I dispositivi HID (human interface device) come ad esempio mouse, tastiere e tavolette grafiche. Il mikrobasic dispone di una libreria per interfacciare al microcontrollore proprio tali periferiche (caratteristica davvero unica per un compilatore di questa classe).

Udp Terminal
È un terminale che permette la comunicazione direttamente con la rete LAN, inviando e ricevendo caratteri. Occorre fornire naturalmente l'indirizzo IP desiderato ed eventualmente il numero della porta di comunicazione. È possibile visualizzare altresì i caratteri inviati nella notazione Ascii, decimale ed esadecimale. Anche questa è una funzione davvero unica.

Mmc Terminal
Si tratta di un utile terminale che consente l'interfacciamento alle nuove memorie flash MMC (Multi Media Card). La comunicazione avviene attraverso la porta seriale del PC ed è possibile leggere o scrivere interi dati da queste memorie ormai di largo uso. Il collegamento al computer avviene attraverso un apposito adattatore. Questo tipo di memorie ormai è utilizzato in tutti i dispositivi di ultima generazione quali fotocamere, cellulari, lettori MP3, ecc, con spazio di massa fino a 1 GB. Inutile dire che questo apre le porte a tutta una serie di idee di progetto, altrimenti irrealizzabili vista proprio la complessità della gestione di tali memorie, MikroBasic lo rende davvero semplice ed intuitivo.

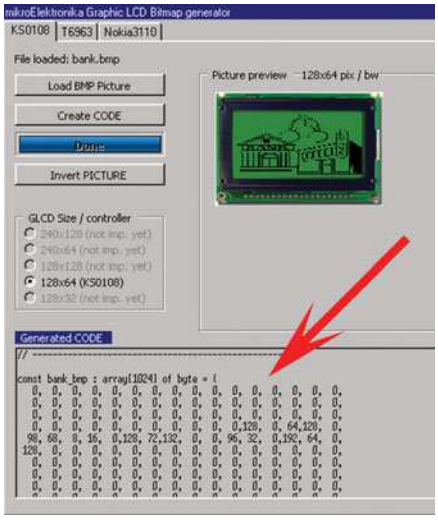


Figura 7 GLCD Bitmap Editor

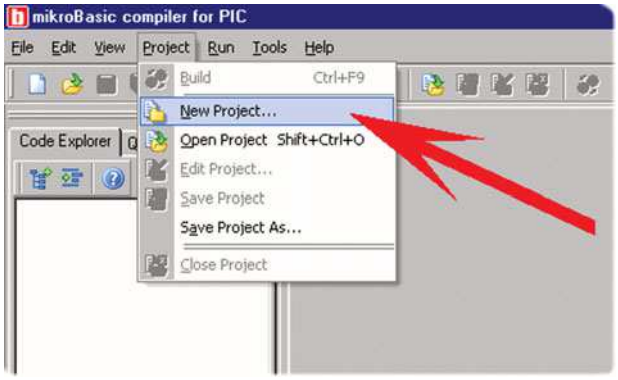


Figura 8 Il menù di creazione Progetto

Glcd Bitmap Editor
Sicuramente uno strumento utilissimo per chi programma ed utilizza I display grafici Lcd (Glcd). Il tool prevede (alla versione attuale) la gestione dei tre tipi più diffusi display:

- KS0108 con matrice bianco/nero di 128 x 64 pixel.
- T6963 con matrice bianco/nero di 128 x 128 pixel.
- Nokia 3110.

Per visualizzare un immagine o un testo su questi tipi di display, occorre convertire le informazioni in sequenze di numeri binari, corrispondenti ai pixel accesi o spenti, e dare infine tale codifica al microcontrollore. Tale preparazione potrebbe essere alquanto lunga e laboriosa, ma grazie a questo potente strumento, basta caricare un file grafico bitmap, dalle appropriate dimensioni, per generare automaticamente il listato basic già pronto contenente la codifica del disegno (figura 7).

CREIAMO UN PROGETTO
Dopo aver visionato le possibilità offerte da questo fantastico compilatore, iniziamo a studiare le diverse fasi della realizzazione di un progetto. Fasi che, ovviamente, molti di voi già conoscono, in special modo chi segue dall'inizio questo corso. La prima fase da seguire è naturalmente la creazione di un nuovo lavoro, appunto di un nuovo progetto. Occorre accedere al menù **Project** e scegliere la voce **New Project** (figura 8). Apparirà una finestra (figura 9) che invita l'utente a fornire dati e indicazioni generali sul tipo di progetto che si vuole creare:

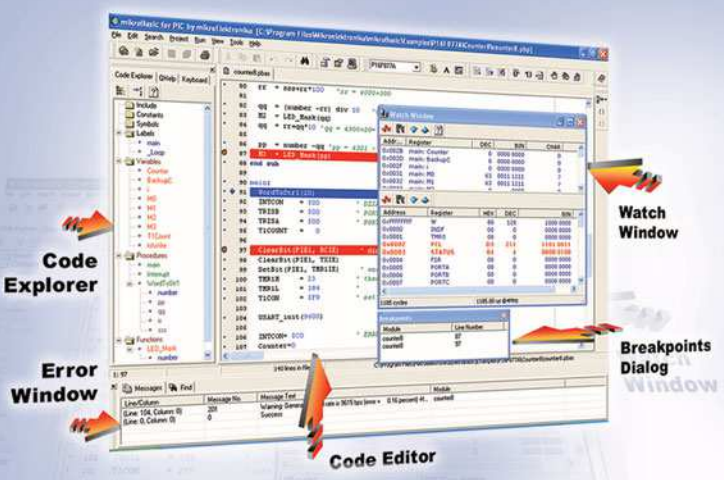
- **Project Name** è il nome che si vuol attribuire al programma e che in definitiva costituisce il nome dei file prodotti.
- **Project Path** è il percorso completo dove il sorgente e gli altri file prodotti saranno memorizzati. Si può selezionare il percorso cliccando sul pulsante **Browse**.
- **Description** è una breve descrizione (opzionale) sul programma che si sta creando, utilissima per tenere traccia delle diverse versioni e delle loro peculiarità.
- **Device** è il tipo di Microcontrollore utilizzato. Nel nostro caso si scelga il P16F84A.
- **Clock**: è la velocità del quarzo che utilizzeremo nel progetto (espresso in Mhz).

Per quanto riguarda i Flag, occorre "puntare" quello relativo alla voce **WDT_OFF** e quello relativo alla voce **XT_OSC**.
Infine occorre confermare le proprie scelte cliccando il pulsante **OK**. Si è adesso pronti a scrivere il codice Basic del programma.

SCRIVIAMO IL PROGRAMMA

Nelle precedenti puntate si è ampiamente approfondito tale aspetto. Si tratta infatti di scrivere nella finestra dell'editor il programma con una corretta codifica del linguaggio. In figura 10 ne riportiamo un esempio. Si tratta di uno dei primi programmi incontrati nel corso. Una volta scritto il codice, si può lanciare la *compilazione* del sorgente.
Tale procedura effettuerà una serie di operazioni per la produzione del file HEX che sarà memorizzato all'interno del PIC. Per compilare il programma occorre cliccare sulla barra degli strumenti il pulsante che evidenzia due ingranaggi gialli, oppure sul menù **Project** e quindi sulla voce **Build**.
Il compilatore inizierà il lavoro della compilazione e, se tutto è a posto, produrrà il file HEX. Lo stato di tale processo è visualizzato da una *Progress-Bar* (figura 11).
Se il codice è stato scritto correttamente, il successo sarà testimoniato dalla finestra dei

Software Mikrobasic



Un potente compilatore Basic per PICmicro

✔ Code Editor
✔ Code Explorer
✔ Debugger
✔ Statistiche

Tutto in un ambiente Windows facile ed intuitivo

Un set di strumenti veramente indispensabili per sviluppare applicazioni con i PICmicro

Ordinalo subito su www.farelettronica.com oppure telefona allo 02.66504794

messaggi, come mostra la figura 12. Invece, nel caso in cui il sorgente sia affetto da errori di qualsiasi genere (e ciò accadrà molto spesso credetemi…) la finestra dei messaggi sarà un tantino più "severa": nessun file sarà generato e compariranno dei messaggi riportanti il tipo di errore, magari logico o di sintassi. Tali messaggi saranno sempre visualizzati nella stessa finestra, come visibile in figura 13.

I FILE GENERATI

Se la compilazione ed il linking del sorgente hanno successo, il MikroBasic genera, nella cartella di lavoro, alcuni file con estensioni particolari, andiamo a descriverli:

- **.ASM** è un file Ascii contenente il sorgente codificato in Assembler. Notare che tale codifica è eseguita totalmente dal compilatore, quindi non dispone di alcuna ottimizzazione e, in generale, non è efficiente come quello creato dall'uomo.
- **.HEX** è il file più importante, in quanto rappresenta quello che deve essere "copiato" (attraverso il programmatore hardware) all'interno del microcontrollore PIC.
- **.LST** è un file di testo che contiene un report di riepilogo dell'allocazione della memoria, degli indirizzi, dei registri, delle routine e delle etichette utilizzate nel sorgente;
- **.MCL** è un file binario contenente l'applicazione creata, che può essere linkata ad altre applicazioni.
- **.PBAS** è il file che effettivamente contiene il sorgente scritto in linguaggio Basic.
- **.PBP** è un file di testo che raccoglie le informazioni circa i dati principali della definizione del progetto (vedi figura 9).

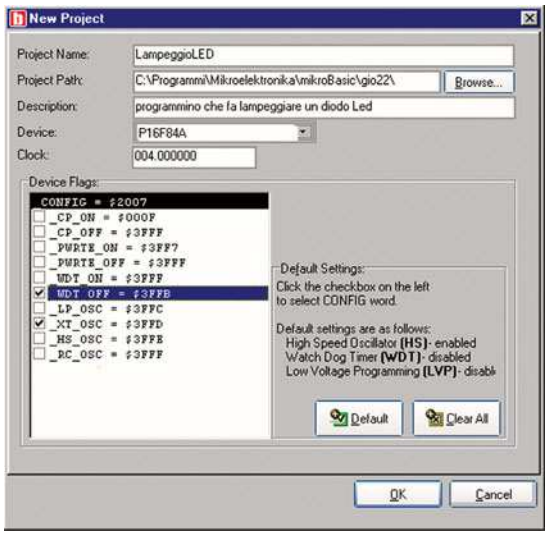


Figura 9 *Informazioni basilari del progetto*

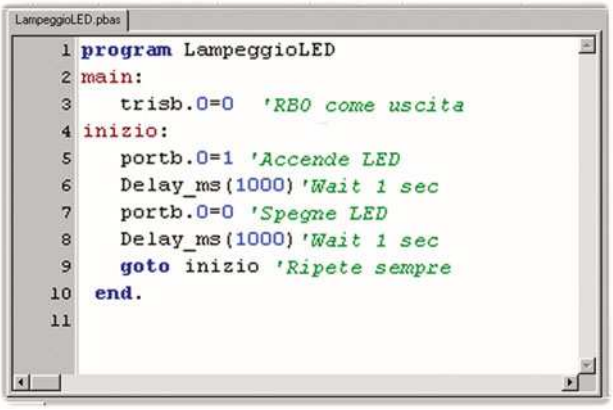


Figura 10 *L'Editor con un programma corretto*

IL DEBUG

Il debugger è uno dei tool più sofisticati e richiesti dai programmatori. È un mezzo estremamente potente con cui si simula, nel PC, l'esecuzione del programma creato, consentendone il controllo totale. Ecco in sintesi le possibilità offerte da un debugger:

- Iniziare e interrompere l'esecuzione del programma a piacimento.
- Inserire dei *break point* (punti di interruzione) nel codice.
- Comandare l'esecuzione del programma passo a passo.
- Ispezionare le variabili contestualmente all'esecuzione.
- Modificare il valore delle variabili senza bloccare la procedura.
- Variare il corso del programma senza alterarne la struttura.

Un debugger è come usare un microscopio che ci consente di esaminare a livello di "bit" il programma scritto, onde scoprire errori, malfunzionamenti e imprecisioni in generale.
La cosa più interessante è che il debugger del Mikrobasic avviene a livello di codice sorgente, nel senso che il controllo è eseguito sul programma Basic e non sull'eseguibile prodotto; questo per permettere una maggiore flessibilità nel controllo e nella verifica. È importante

sottolineare che il processo di debug può essere correttamente eseguito solo dopo che la compilazione è stata conclusa con successo.
Dopo aver scritto il proprio sorgente e averlo correttamente compilato, si può passare direttamente al debug del codice, non è una fase obbligatoria ma la si può utilmente utilizzare quando il programma non fornisce i risultati desiderati. Per eseguire il debugger occorre premere il tasto **F9** oppure accedere al menù **Run** e scegliere la voce **Start Debugger**, si aprirà la finestra *Watch* (figura 14). Come si può notare, in essa sono riportate le *variabili* usate nel programma (le variabili *utente* sono evidenziate dalla parentesi graffa), gli indirizzi della loro locazione e, soprattutto, il valore corrente, in notazione *esadecimale*.
Bisogna considerare un aspetto importantissimo: il debugger simula il flusso del programma e l'esecuzione della varie linee di istruzioni, ma **non** emula esattamente il comportamento del PIC. Vediamo in dettaglio i tasti da utilizzare per controllare le diverse opzioni:

- **F9** inizia la procedura di debug.
- **F6** alternativamente ferma e riavvia il debug.
- **F5** posiziona un break point alla posizione del cursore; se è già presente, questi è annullato.
- **F4** esegue tutte le istruzione dall'ultima espletata fino alla posizione del cursore.
- **F7** (Step Into) esegue passo per passo le istruzioni. Se è chiamata una funzione o procedura, esegue anch'essa passo per passo, mostrando la scansione dell'esecuzione. Utilissima per controllare il flusso in una UDF.
- **F8** (Step Over) esegue passo per passo le istruzioni. A differenza del precedente *Step Into*, l'eventuale esecuzione di una funzione o procedura avviene in maniera nascosta.
- **CTRL+F8** (Step Out) esegue completamente il programma sino alla fine.
- **CTRL+F2** (Stop Debugger) termina il processo di debug.

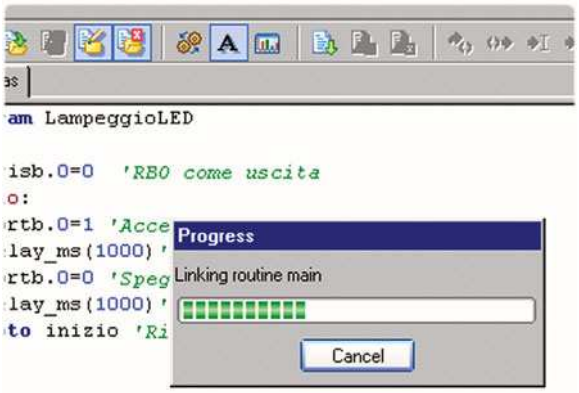


Figura 11 Compilazione in corso

Manipoliamo le variabili

L'esecuzione passo per passo permette di "sbirciare" all'interno di una variabile, con la possibilità di cambiarne il valore, vediamo come operare.
Durante l'uso dello *Step Into* oppure dello *Step Over*, i valori delle variabili sono aggiornati "in tempo reale" e visualizzati nella stessa finestra. Se occorre modificare il valore di una variabile (ad esempio per mutare il corso di un *ciclo* oppure per ribaltare il risultato di un *flag*) baste-

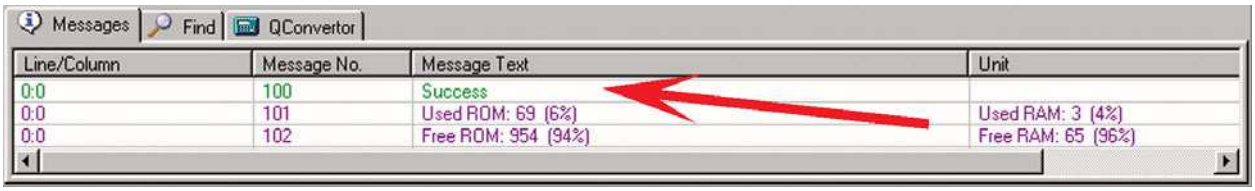


Figura 12 Compilazione eseguita con successo

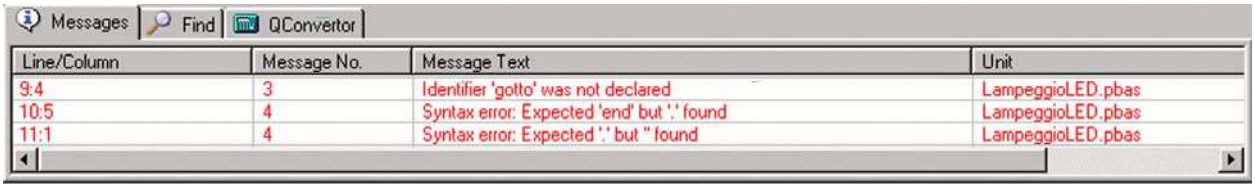


Figura 13 Compilazione interrotta per la presenza di un errore di programmazione

rà fare doppio click sul nome della stessa variabile all'interno della *Watch Window*. Pertanto una nuova finestra (Edit Value ossia Modifica Valore) inviterà l'operatore ad inserire un nuovo valore, che andrà subito ad aggiornare la variabile. Il parametro da modificare potrà essere immesso nella notazione binaria, decimale, esadecimale, float ed ascii (figura 15). Si ha altresì modo di scegliere la modalità di visualizzazione sulla *Watch Window* (binario, esadecimale, carattere, decimale, float).

Lo Stopwatch

Durante l'esecuzione "virtuale" del programma sviluppato, quindi nella fase del debug, è possibile *monitorare* un'altra famiglia di informazioni.
Nella Watch Window (attiva durante il debug), alla pressione della cartella *Stopwatch*, si accede ad un insieme di informazioni, relative soprattutto all'andamento *temporale* del flusso del programma scritto. Come si vede dalla figura 16, durante l'esecuzione del software in Basic in modalità *step by step* (con la pressione dei tasti F7 oppure F8) alcuni "contatori" sono continuamente aggiornati.
In pratica essa mostra il totale dei cicli (passi o step) eseguiti dal microcontrollore ed il relativo tempo trascorso (tempo virtuale – reale) dall'avvio del debugger, espresso in microsecondi (milionesimi di secondo). Il campo *delta* mostra invece il numero dei cicli (e relativi tempi) che intercorrono tra la *precedente* istruzione che il debugger ha eseguito e l'istruzione *attiva* (selezionata da una barra di colore blu) che deve essere ancora eseguita.

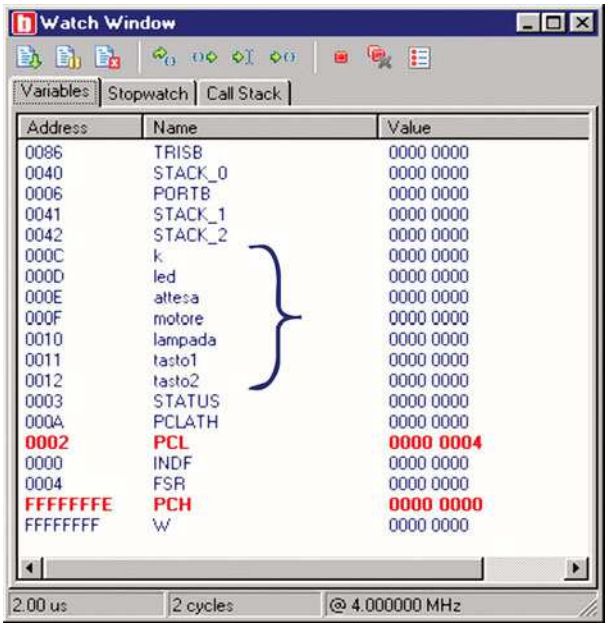


Figura 14 Finestra Watch

CONCLUSIONI

Bene, siamo arrivati anche questa volta al termine della puntata. L'esperienza è, come sempre, la migliore maestra, quindi datevi sotto con le sperimentazioni perché, come si sa, sbagliando si impara.
Dopo un lungo utilizzo, anche l'ambiente del compilatore del Mikrobasic, diventerà familiare e, pian piano, si troveranno anche trucchi e metodi per "ottenere" di più.
Vi rinnovo l'appuntamento alle prossime puntate con altre sperimentazioni divertenti, utili ed interessanti, al fine di ottenere sempre i massimi risultati dai nostri progetti in MikroBasic.

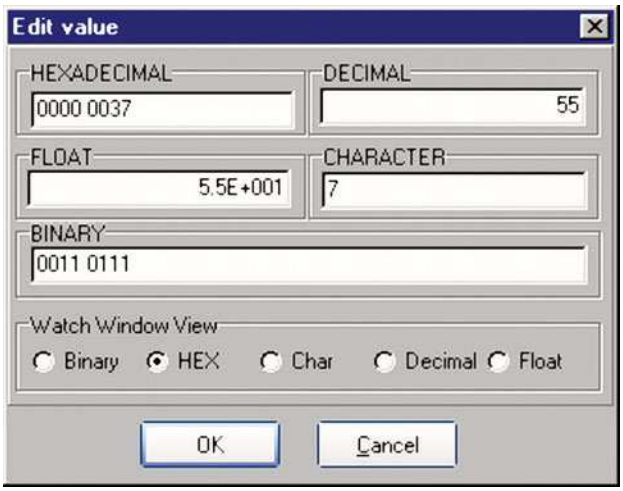


Figura 15 Finestra Edit Value

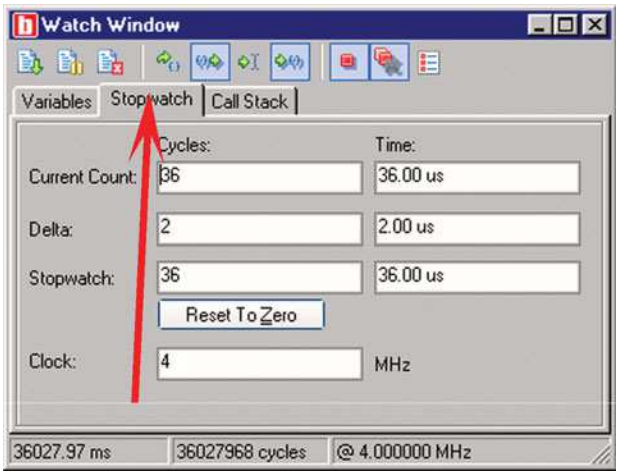


Figura 16 Le informazioni temporali dello Stopwatch

con il patrocinio del
Comune di Rimini

Expo Elettronica® 2006

mostra mercato

elettronica • hardware • software
telefonia • materiali di consumo
hobbistica • accessori • componenti
surplus • ricezione satellitare

Vieni a conoscere LINUX!
Dimostrazioni e mini-conferenze sul software libero
a cura di RiminiLUG (www.riminilug.it)

NUOVA EDIZIONE! 150 espositori di elettronica & Co.

Palacongressi Riviera di Rimini

RIMINI
4·5 febbraio
ore 9/18

(vecchia fiera) Via della Fiera, 52 (Uscita A14 - Rimini Sud)

rif. FARE ELETTRONICA

RADIO EXPO'

Mostra Mercato del Radioamatore

Saranno presenti in fiera le principali ditte del settore:

IRAE · arno elettronica · milag · GR Pubblicazioni · RADIO SYSTEM · ECO ANTENNE

Sponsor Expo Elettronica 2006
www.mybay.it
myBAY
BBS · www.9and99.com · www.snailweb.it
sponsored by somiwo
www.somiwo.com

organizzazione: BLU NAUTILUS • tel. 0541 439573 • www.blunautilus.it

Per ottenere un INGRESSO RIDOTTO scarica il biglietto dal sito www.blunautilus.it o presenta questa inserzione alla cassa

gli appuntamenti

03-04 Dicembre 2005

GRANDE FIERA DELL'ELETTRONICA — Forlì (FC)

Alla Fiera di Forlì appuntamento con uno dei più importanti eventi nazionali "consumer" di settore; quello con la Grande Fiera dell'Elettronica. Le cifre sono quelle delle grandi occasioni: oltre 350 espositori, 20.000 mq di esposizione e più di 20.000 visitatori attesi alla manifestazione. La Grande Fiera dell'Elettronica ha un pubblico vasto ed eterogeneo: appassionati del "fai da te", elettro-riparatori, "smanettoni", radioamatori, "cacciatori" di buone occasioni o pezzi rari; questo perché propone un panorama merceologico e un calendario di eventi collaterali veramente ricchissimo. Al suo interno si svolgono diversi saloni tematici: La Fiera dell'Astronomia amatoriale, Photo cine video, Radio Expo mercatino delle radiocomunicazioni, Dischi e cd da collezione e il concorso nazionale dell'inventore elettrico-elettronico.

Luogo: Fiera di Forlì - Forlì (FC)
Orari: Dalle 9:00 alle 18:00
Organizzatore: Blu Nautilus srl (www.blunautilus.it - Tel 0541.439573)
Ingresso: € 8,00 (ridotto € 7,00)

10-11 Dicembre 2005

MOSTRA MERCATO NAZIONALE RADIANTISTICA ELETTRONICA — Civitanova Marche (MC)

Mostra Mercato Nazionale della Radiantistica Elettronica: materiale radiantistico per C.B. e radioamatori, apparecchiature per telecomunicazioni, surplus, telefonia, computers, Antenne e parabole per radioamatori e Tv Sat, radio d'epoca, editoria specializzata, Play-Station - videogiochi – flipper. Disco: mostra mercato del disco usato in vinile e CD da collezione.

Luogo: Quartiere Fieristico – Civitanova Marche (MC)
Orari: Dalle 9:00 alle 13:00 e dalle 15:00 alle 19:30
Organizzatore: ERF (www.erf.it - Tel 0733.780811)
Ingresso: n.p.

10-11 Dicembre 2005

COMPUTER FEST & RADIOAMATORE — Torreano di Martignacco (UD)

Torna, nei nuovissimi padiglioni del quartiere fieristico udinese di Torreano di Martignacco, la grande festa dedicata all'elettronica, all' informatica e ai radioapparati. Torna con tantissimi espositori e una grande proposito: riportare agli appassionati della zona un appuntamento annuale a costi d'ingresso accettabili e con parcheggio gratuito.
Per tutti i visitatori dell'edizione estiva di Cerea, ancora muniti di biglietto, l'ingresso sarà completamente gratuito. La fiera sarà articolata su due padiglioni e al suo interno offrirà numerosi servizi.

Luogo: Quartiere fieristico udinese Torreano di Martignacco (UD)
Orari: Dalle 9:00 alle 18:00
Organizzatore: Compendio Fiere S.r.l. (www.compendiofiere.it - Tel. 337.676719)
Ingresso: € 6,00 (ridotto € 4,00)

di DICEMBRE duemilacinque

Per sapere a quali appuntamenti sarà presente fare elettronica visita la pagina: www.farelettronica.com/fiere

10-11 Dicembre 2005

33° TELERADIO & COLLEZIONI — Località Le Mose (PC)

La novità rappresentata da questa nuova edizione consiste nella scelta di rendere biannuale l'appuntamento piacentino: da quest'anno, oltre alla tradizionale edizione di settembre di Teleradio, data fissa e consolidata nel fitto calendario delle manifestazioni fieristiche dedicate a questo settore, ha trovato opportuna collocazione anche un'edizione pre-natalizia, programmata il 10 e 11 dicembre. Un'occasione ghiotta per gli appassionati del mercato secondario del settore dell'elettronica (surplus, elettronica, accessori). Sono previste alcune aree di collezionismo: dischi usati, libri, fumetti e giocattoli usati e mille altre proposte per il collezionismo.

Luogo: Quartiere Fieristico di Piacenza
Località Le Mose (PC)
Orari: Dalle 9:00 alle 18:00
Organizzatore: Piacenza Expo S.p.a.
(www.teleradio.piacenzaexpo.it - Tel 0423.602711)
Ingresso: € 6,00

17-18 Dicembre 2005

25° MARC — Genova

Mostra mercato di attrezzature radioamatoriali, componentistica, hardware, software, ricezione satellitare, editoria specializzata, radio d'epoca.

Luogo: Fiera Di Genova - Genova
Orari: Dalle 9:00 alle 18:30
Organizzatore: Studio Fulcro
(www.studio-fulcro.it - Tel 010.561111)
Ingresso: n.p.

08 Gennaio 2006

ELETTRONICA EXPO DAY — NOVI LIGURE (AL)

Una manifestazione dedicata all'informatica ed al mondo del computer. Tanta tecnologia alla portata di tutte le tasche: telefonia, antenne e decoder per la tv digitale, lettori dvd, hi-fi, videoproiettori, videogiochi; articoli, ricambi, gadget e curiosità elettriche, elettroniche e digitali. Una mostra mercato in cui sarà possibile trovare buone occasioni sia nel "nuovo" sia tra i "surplus", giacenze o usato perfettamente funzionante.
Un appuntamento davvero imperdibile anche per il pubblico più specializzato, che qui trova materiali e componenti molto tecnici e rari da reperire, elementi indispensabili per l'autocostruzione, per personalizzare o riparare tutto ciò che è elettrico o elettronico.
I riparatori poi troveranno quei componenti e pezzi di ricambio indispensabili per riparare l'hi-fi o la tv di qualche anno fa, difficilmente reperibili attraverso i normali canali distributivi. Si potranno trovare anche apparecchi per radioamatori, radio d'epoca, militari, surplus, valvole, ricambi e riviste.

Luogo: Centro Fieristico "Dolci Terre di Novi"
Novi Ligure (AL)
Orari: Dalle 9:00 alle 18:30
Organizzatore: Expofiere
(www.expofiere.com - Tel 333.6252325)
Ingresso: n.p.

Biofeedback

Qui vedremo come trasformare un semplice monitor di battiti cardiaci (presentato nel numero 241 di Fare Elettronica) in uno strumento più raffinato e fedele, grazie ad un filtro in ingresso di insolita concezione che aumenta molto il rapporto segnale/disturbo. Il segnale ECG sarà inoltre trasmesso via radio tramite opportuno modulatore e demodulatore FM. Completano il tutto una breve trattazione del funzionamento del cuore, il significato di alcune anomalie del tracciato ECG, e qualche interessante spunto per nuove ricerche, unitamente a brevi note matematiche.

I FILTRI ED IL TRATTAMENTO DEL SEGNALE

In questo contesto si intende come azione di filtro l'estrazione di un segnale utile nel caso in cui esso sia miscelato ad altro di natura indesiderata: *il rumore*. Questo è un tema ben più affascinante e complesso di quanto possa apparire a prima vista, con implicazioni che si estendono alla teoria dell'informazione, alle scienze sociali, alla filosofia, alla vita quotidiana. Infatti è estrazione del segnale dal rumore la ricerca dati su internet, la lettura oculata di un giornale, la scoperta di una traccia occultata da abili manipolatori dell'informazione. Si possono fare molti altri paralleli ed analogie che lascio all'immaginazione di ognuno. Per quanto riguarda l'estrazione del segnale desiderato vigono però alcuni principi base comuni a tutte le problematiche esposte sopra:

- ***Usare il sensore migliore possibile***, cioè il più adatto allo scopo,quello che genera meno falsi segnali, quello dotato di meno rumore intrinseco. Per un ECG si userà un elettrodo Ag-Cl che minimizza la tensione di polarizzazione; per rivelare deboli campi magnetici (Magnetoencefalografia) un sensore ad effetto quantistico, uno SQUID raffreddato in elio liquido. Per ricercare una informazione si affineranno le proprie capacità di attenzione e concentrazione.
- ***Usare tecniche di filtro***: esse variano in funzione del tipo di rumore che accompagna il segnale e possono essere in tempo reale oppure mediate in un arco di tempo.

Esempio 1: L'informazione che mi interessa ha una banda in frequenza limitata (ECG, EEG), uso dei filtri passa banda che lasciano transitare ciò che è utile e tagliano ciò che non serve.

Esempio 2: il segnale è molto piccolo,ripetitivo, ed è sommerso da rumore, il cui valore medio è nullo o quasi nell'arco di tempo occorrente ad effettuare la misura: in questo caso si effettuano più campionamenti (centinaia, migliaia) del medesimo, tenendo conto del suo periodo di ripetizione; sommando e mediando i vari campionamenti l'informazione che cerchiamo si rafforza mentre il rumore di origine casuale diminuisce – si dice che il rumore non è correlato con se stesso- in questo modo di possono rivelare i potenziali nascosti nell'ECG, nell'EEG, ecc. Con la stessa tecnica astronomi dilettanti, semplicemente mediando tramite computer diversi frame di un pianeta ottenuti tramite una econo-

Trasmissione a distanza

di Vittorio Marradi
vittorioaldo.marradi@fastwebnet.it

mica telecamera ed un telescopio, possono ottenere immagini di buona risoluzione perché i disturbi casuali contenuti nelle varie immagini tendono ad annullarsi. A livello di informazione il parallelo è vagliare e mediare informazioni ottenute da fonti diverse, possibilmente con interessi politici ed economici contrastanti.

Esempio 3: Il rumore è di natura particolare e prevedibile. In questa situazione si studiano provvedimenti che tengono conto della natura dello stesso: nel caso di ECG, EEG il rumore di modo comune viene molto attenuato usando un amplificatore differenziale ad alto CMRR. Altro esempio per disturbi della stessa natura: se masterizziamo tramite apposito software un disco in vinile rigato, possiamo ridurre fortemente i toc-toc della riproduzione analizzando lo spettro in frequenza del disturbo (tramite analisi di Fourier veloce o FFT) e sottraendolo con tecniche digitali al segnale complessivo. Per fare un altro parallelo si può notare che conoscere le motivazioni per le quali un'informazione viene distorta – vecchio adagio cinese: guarda chi ha un utile e scoprirai il mandante – cioè fare un'analisi dell'azione di disturbo e sottrarre il movente alla propaganda, consente di arrivare vicino al nocciolo di verità.

Quando il segnale è piccolo occorre ***usare l'amplificatore più fedele possibile***, cioè quello con la banda passante più alta compatibilmente con le esigenze di soppressione del rumore o,in altre parole,quello che conserva meglio le componenti dello spettro di Fourier dell'informazione originale. Nel caso dell'ECG ad esempio, un filtro passa basso e passa alto, non correttamente progettati, portano ad una alterazione delle costanti di tempo del circuito, con modifiche del tracciato e conseguenti diagnosi erronee. Per scongiurare ciò esistono normative internazionali che prescrivono la banda passante degli amplificatori biomedicali cardiaci tipicamente da 0.05Hz fino a 150 Hz.
Inevitabilmente si può notare che anche nel campo dell'informazione la diffusione di verità parziali (si annullano alcune componenti…) porta spesso l'utente delle medesime a valutazioni e conclusioni non corrette. Nel caso del nostro amplificatore occorre pertanto migliorare, al fine di avere un buon segnale, la risposta dei filtri passa-alto e passa-basso, rispettivamente del primo e del secondo ordine.
Come funzionano i filtri? Ecco un PB, figura 1 ed un PA figura 2 ambedue del 1 ordine: la loro risposta in frequenza è descritta dalla funzione di trasferimento Vu/Vi che si ricava semplicemente dalla legge di Ohm, tenendo conto che l'equivalente della resistenza in un circuito in corrente alternata si chiama reattanza e vale, per un condensatore e in forma del tutto generale, 1/sC dove s è la trasformata di Laplace dell'alternanza in ingresso.
La forma d'onda può essere infatti triangolare, quadra, a dente di sega ecc. ad ognuna di queste corrisponde una diversa s-trasformata, o trasformata di Laplace (fine 800). Nel caso particolare di correnti alternate sinusoidali la s-trasformata si semplifica nella notazione $j\omega$ e la reattanza del condensatore diventa $1/j\omega C$ dove l'operatore $j = \sqrt{-1}$. Questo operatore, che implica particolari regole di moltiplicazione, divisione ecc. - viene adoperato per effettuare

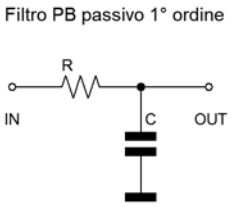


Figura 1 Filtro passa basso

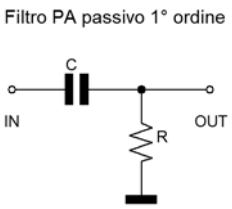


Figura 2 Filtro passa alto

operazioni fra vettori nel cosiddetto piano complesso o di Gauss. Anche questa è matematica del 700/800 che al giorno d'oggi, in un paese di poeti, navigatori, santi, si insegna a studenti particolarmente ricettivi di elettrotecnica o elettronica nelle scuole superiori.

UNA BREVE ANALISI MATEMATICA

Si chiama funzione di trasferimento Ft di un filtro il rapporto tra tensione di uscita e di ingresso dello stesso. Nel caso sinusoidale ed omettendo per semplicità la notazione vettoriale, in assenza di assorbimento all'uscita si semplifica la corrente al numeratore e denominatore e abbiamo che la Ft è:

$$\frac{Vu}{Vi} = \frac{1/j\omega C}{R+(1/j\omega C)} = \frac{1}{J\omega RC+1}$$

per un filtro PB del 1° ordine. Quando i moduli delle due componenti al denominatore sono eguali cioè $\omega RC = 1$ si dice che siamo in presenza di un polo dell'equazione, un cambiamento di pendenza da orizzontale verso il basso della Ft per ω crescenti e la frequenza di taglio vale

$$f = \frac{1}{2\pi RC}$$

essendo $\omega = 2\pi f$.In queste condizioni l'attenuazione del filtro è in modulo $1/\sqrt{2} = 0.707$ pari a -3 dB. Per il PA la Ft è,con le medesime ipotesi di prima:

$$\frac{Vu}{Vi} = \frac{J\omega RC}{J\omega RC+1}$$

stessa soluzione per la frequenza di taglio. La differenza sta nell'influenza del numeratore per il quale si dice che esiste uno zero passante per l'origine. Questo comporta in parole povere che la Ft sale per ω crescente, e diventa orizzontale nei dintorni della frequenza di taglio.

Un filtro del secondo ordine PB, quello usato nell'ECG ha una trattazione più complessa, la sua Ft, in forma generale in notazione di Laplace è

$$= \frac{A_0\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2}$$

Viene utile qui fare un parallelo: l'equazione al denominatore è infatti analoga a quella che descrive un sistema oscillante smorzato, come ad esempio gli ammortizzatori di un veicolo.
Essi sono costituiti da una molla (elemento elastico – oscillante) ed un insieme cilindro - pistone che trafila olio (elemento smorzatore); queste due componenti devono essere bilanciate per un assorbimento ottimale delle asperità del terreno e rappresentano un filtro PB fra queste sollecitazioni e le oscillazioni del veicolo. Nell'equazione vista prima abbiamo anche che $Q = 1/2\zeta$. Il termine ζ che rappresenta lo smorzamento ed è inversamente proporzionale al termine Q (fattore di merito), che è una funzione delle qualità di oscillazione del dispositivo, sia esso meccanico od elettronico. È proprio la scelta di ζ che condiziona l'azione di taglio del filtro, che può essere più o meno rapida ed essere affetta o meno da sovra-oscillazioni. Alcune fra le soluzioni all'equazione per diversi valori di ζ sono i particolari polinomi di Bessel, di Butterworth, di Chebyshev: il filtro è sempre lo stesso, variano opportunamente i rapporti fra le resistenze ed i condensatori che lo costituiscono a seconda del tipo di soluzione scelta.
Ad esempio il polinomio di Butterworth normalizzato ad $\omega_0 = 1$ rad/s di secondo grado è $s^2 + \sqrt{2}s + 1$. Il polinomio di 2° grado di Bessel s^2+3s+3. Quello di Chebyshev con ripple di sovra-oscillazione di 0.5dB è $s2+1.425624s+1.516203$.
Questi polinomi sono del tutto generali ed usati anche per altre applicazioni, ad esempio collegando una rete di altoparlanti in serie- parallelo con opportune fasi, secondo una disposizione ricavata dai coefficienti di un polinomio di Bessel è possibile ricavare un fronte d'onda sonoro direzionale.
La trattazione completa di questo argomento richiederebbe interi capitoli, imponendo però alcune condizioni si semplifica enormemente il problema. Infatti utilizzando un operazionale con guadagno 1, detto voltage follower e la soluzione di Butterworth ed imponendo la coppia di resistenze eguali si ha che la pulsazione naturale ω_0, dovuta ai gruppi RC, coincide con quella di taglio a -3dB, inoltre $Q = \zeta = \sqrt{2}/2 = 0.707$ per cui $2\zeta = \sqrt{2}$, In queste condizioni risulta assente il picco di sovra-oscillazione. È semplificato il calcolo dei componenti in quanto la Ft diventa

$$\frac{1/R^2C_1C_2}{s^2 + 2s/RC_1 + 1/R^2C_1C_2}$$

Una serie completa di scatole di montaggio hi-tech che sfruttano la rete GSM.

GSM SOLUTIONS

APRICANCELLO

Facilmente abbinabile a qualsiasi cancello automatico. Attiva un relè di uscita (da collegare all'impianto esistente) quando viene chiamato da un telefono fisso o mobile precedentemente abilitato. Programmazione remota mediante SMS con password di accesso. Completo di contenitore e antenna bibanda. Alimentatore non compreso.

FT503K Euro 240,00

TELECONTROLLO

Sistema di controllo remoto che consente di attivare, mediante normali SMS, più uscite, di verificare lo stato delle stesse, di leggere il valore logico assunto dagli ingressi nonché di impostare questi ultimi come input di allarme. Possibilità di espandere gli ingressi e le uscite digitali. Funziona anche come apricancello. Completo di contenitore.

FT512K Euro 255,00

TELEALLARME A DUE INGRESSI

Invia ad uno o più utenti un SMS di allarme quando almeno uno degli ingressi viene attivato con una tensione o con un contatto. Può essere facilmente collegato ad impianti di allarme fissi o mobili. Ingressi fotoaccoppiati, dimensioni ridotte, completamente programmabile a distanza.

FT518K Euro 215,00

CONTROLLO REMOTO 2 CANALI CON TONI DTMF

Telecontrollo DTMF funzionante con la rete GSM. Questa particolarità consente al nostro dispositivo di operare ovunque, anche dove non è presente una linea telefonica fissa. Può essere chiamato e controllato sia mediante un cellulare che tramite un telefono fisso. Il kit comprende il contenitore; non sono compresi l'antenna e l'alimentatore.

FT575K Euro 240,00

ASCOLTO AMBIENTALE

Sistema di ridotte dimensioni per l'ascolto ambientale. Può essere facilmente nascosto all'interno di una vettura o utilizzato in qualsiasi altro ambiente. Regolazione della sensibilità da remoto, chiamata di allarme mediante sensore di movimento, password di accesso. Viene fornito con l'antenna a stilo, mentre il sensore di movimento è disponibile separatamente.

FT507K Euro 280,00

MICROSPIA TELEFONICA

Collegata ad una linea telefonica fissa, consente di ascoltare da remoto tutte le telefonate effettuate da quella utenza. La ritrasmissione a distanza delle telefonate sfrutta la rete GSM. Microfono ambientale supplementare, I/O a relè. La scatola di montaggio non comprende il contenitore e l'antenna GSM.

FT556K Euro 245,00

COMMUTATORE TELEFONICO

Collegato al telefono di casa effettua automaticamente una connessione GSM tutte le volte che componiamo il numero di un telefonino. In questo modo possiamo limitare il costo della bolletta in quanto una chiamata cellulare-cellulare costa quasi la metà rispetto ad una chiamata cellulare-fisso. Il kit non comprende il contenitore e l'antenna GSM.

FT565K Euro 255,00

FUTURA ELETTRONICA

Via Adige, 11 -21013 Gallarate (VA)
Tel. 0331/799775 - Fax. 0331/778112 - www.futuranet.it

Maggiori informazioni su questi prodotti e su tutte le altre apparecchiature distribuite sono disponibili sul sito www.futuranet.it tramite il quale è anche possibile effettuare acquisti on-line.

Tutti i prezzi si intendono IVA inclusa.

dove C1 è il condensatore verso massa C2 quello di reazione. Quindi

$$\omega_0 = \sqrt{1/R^2 C_1 C_2}$$

da cui la frequenza di taglio

$$f_t = 1/2\pi R\sqrt{C_1 C_2}$$

Lo smorzamento deve essere

$$\zeta = \sqrt{C_2/C_1} = 0{,}707$$

il che si verifica quando C2 = 2C1.
Che filtro scegliere per il nostro ECG? Ovviamente ogni soluzione ha pregi e difetti: con Butterworth la risposta in banda passante è piatta, ma la risposta in fase non è molto lineare. Con Bessel migliora la risposta in fase ma l'azione di taglio è molto più graduale, si ha un maggiore ζ. Per curiosità scaricando gratuitamente la versione limitata gratuita di SPICE dal sito e facendo clik sinistro su Lpfilter.ckt seguita da F10 si può osservare la risposta in frequenza (diagramma di Bode) per i tre casi menzionati cioè Bessel, Butterworth e Chebyshev cambiando il valore del condensatore di reazione.

IL CUORE

Lungi da me l'idea di sostituirmi all'arte medica, o anche solo quella di dare indicazioni, le note che seguono sono allo scopo di soddisfare qualche curiosità. ***Il cuore***: questo potente muscolo, il cui spaccato si vede in (figura 3) pompa in una sua sezione sangue povero di ossigeno nei polmoni, dove poi i globuli rossi si arricchiranno di questo gas rilasciando anidride carbonica, prodotto della combustione metabolica.
Nell'altra sua sezione spinge il sangue ossigenato nel corpo.
Tutto ciò viene fatto mediante quattro cavità, le cui contrazioni e dilatazioni esercitano l'azione pompante, ed un sistema opportuno di valvole. Queste azioni devono essere perfettamente sincronizzate da una precisa temporizzazione degli impulsi elettrici che azionano questi comparti.
Ciò che possiamo osservare con l'ECG è una media degli impulsi elettrici associati agli eventi di dilatazione e contrazione, in quanto essi vengono raccolti a livello della pelle e, nonostante questa limitazione, la forma degli impulsi e gli intervalli di tempo fra essi forniscono utili indicazioni.

Le cellule cardiache possono generare tensioni dell'ordine di 100mV, di polarità negativa o positiva alternando l'equilibrio degli ioni sodio Na+ potassio K+ fra interno ed esterno delle medesime attraverso opportuni canali di conduzione siti sulla membrana. A riposo la cellula è polarizzata negativamente, sotto stimolo si depolarizza,attivandosi e, cambiando il suo stato,assume una polarità positiva.

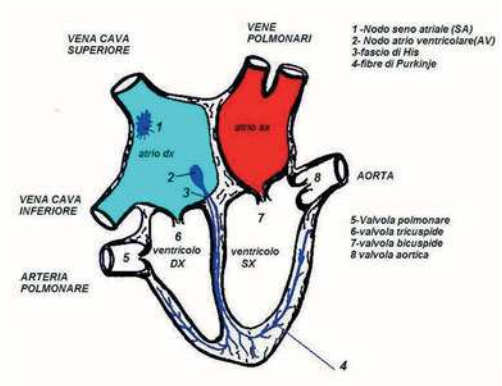


Figura 3 Il cuore

L'impulso di depolarizzazione (attivazione) si propaga attraverso appositi insiemi di cellule (si parla di propagazione di un dipolo elettrico) che sono: nodo AS, nodo AV, fascia di His,le fibre di Purkinje. Questo si verifica analogamente ad un impulso elettrico che percorre una linea di ritardo, formata da filtri RC passa-basso. In pratica il funzionamento globale è comandato da un nodo di eccitazione primario, un clock la cui frequenza dipende dalle condizioni corporee, è il cosiddetto *pacemaker principale*, il segnapassi, il nodo SA. Continuando il parallelo con l'elettronica, questo clock si propaga lungo linee di ritardo con coefficiente di trasmissione temporale diverso in modo da sincronizzare opportunamente i movimenti delle quattro cavità, due atri e due ventricoli. È del tutto ovvio che alterazioni nella conduzione e lesioni del tessuto cardiaco portino come risultato un malfunzionamento del sistema e, a livello diagnostico una modifica del tracciato ECG.
Approfondendo un poco, e con riferimento alla figura 3, abbiamo che dai vari organi il sangue povero di ossigeno confluisce nell'atrio destro, una sorta di precamera, poi, attraverso la valvola tricuspide affluisce nel ventricolo destro, costituito da un più potente tessuto muscolare, il quale contraendosi spinge il sangue nell'arte-

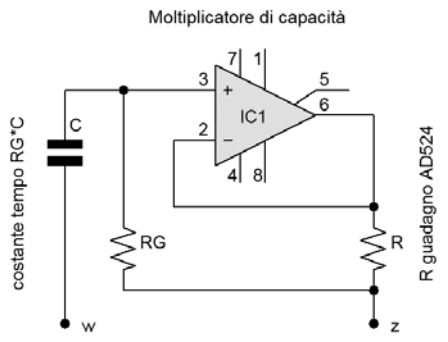


Figura 4 Moltiplicatore di capacità

ria polmonare, attraverso l'omonima valvola. Rifluendo dai polmoni il sangue raggiunge l'atrio sinistro, passa attraverso la valvola mitrale, arriva nel ventricolo sinistro. Questo si contrae e pompa il rosso fluido attraverso la valvola aortica e, tramite l'aorta, nel corpo. L'impulso primario di eccitazione elettrica si forma nel nodo seno-atriale SA attraversa velocemente l'atrio,velocità 1 m/sec, fino al nodo atrioventricolare AV che ritarda la conduzione a 5 cm/sec per permettere ai ventricoli di riempirsi e si distribuisce fino a questi tramite il fascio di His e la rete di Purkinje, 40 cm/sec. L'attivazione (depolarizzazione) degli atri viene rilevata come onda P, dopo un ritardo di circa 20 mSec e la contrazione dei ventricoli forma il complesso QRS (vedere prima parte), che data la grande ampiezza maschera la polarizzazione (disattivazione) degli atri. L'onda T generata dalla fine del lavoro dei ventricoli, chiude il ciclo che viene anche descritto come sistole e diastole.

Alcune anomalie del tracciato

Frequenza cardiaca: misurando il periodo Temp, in millisecondi, fra due picchi R successivi i battiti per minuto sono Frequenza cardiaca= 60000/ Temp. Un confronto rispetto alla norma rivela tachicardia o brachicardia, cuore troppo veloce o troppo lento. Se il tratto PQ > 20 mSec si hanno **disturbi di conduzione** dell'impulso di diversa gravità, dall'allungamento del tratto P-Q, ad onde P a volte non seguite dal complesso QRS, od anche dissociate dal medesimo; si parlerà di **blocchi** atrioventricolari di diverso grado, primo, secondo, eccetera. Ancora, in caso di lesioni, infarto, si avranno alterazioni anche nella componente continua del segnale ECG.

Se **la distribuzione degli impulsi di comando** a due ventricoli non è isocrona si noteranno alterazioni del picco R come: doppia punta, allargamento della cuspide. Concentrazioni alte di calcio possono provocare QT breve, viceversa se basse - anomalie nella concentrazione di potassio alterano la forma dell'onda T infatti cambiano le velocità di migrazione ionica fra interno ed esterno delle cellule. Si possono altresì rivelare ipertrofie, auto-oscillazioni, residui di infarto ed altre patologie legate alla forma stessa degli impulsi ed al fronte di salita e di discesa dei medesimi.

L'osservazione di questi fenomeni dipende, e può essere più o meno evidenziata, dalla disposizione degli elettrodi esploranti.

L'AMPLIFICATORE

Nella prima parte abbiamo usato un filtro PA in ingresso, al fine di limitare i disturbi di polarizzazione dovuti agli elettrodi, direttamente implementato sulla rete che determina il guadagno dell'amplificatore per strumentazione AD 524 e costituito dal condensatore C11 collegato fra RG2 ed il pin di guadagno desiderato. Tale solu-

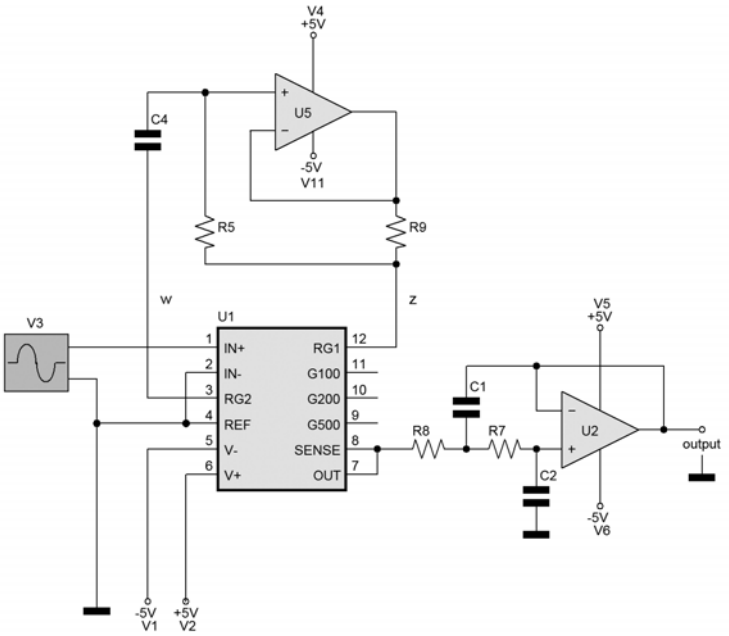


Figura 5 Modello SPICE PA con operazionale

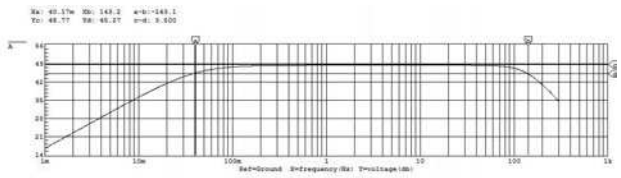


Figura 6 Diagramma di Bode con PA con LF351

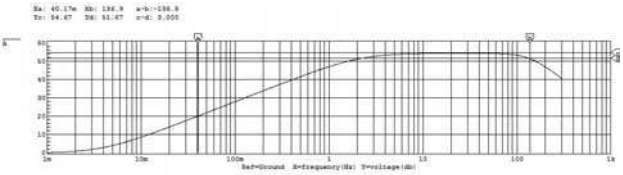


Figura 7 Diagramma di Bode con PA semplice RC

zione era buona per rivelare gli impulsi, ma non per conservarne la forma, la frequenza di taglio PA era troppo alta e occorreva aumentare la capacità C11 ben oltre i 1000 uF. Notiamo che il guadagno può anche essere impostato a piacere collegando una Rg del valore voluto tra Rg1 – Rg2. L'azione di filtro si ottiene inserendo in serie a tale Rg una capacità C. Siamo però quasi al punto di prima in quanto Rg * C determina la costante di tempo del filtro, che deve essere alta, però Rg deve essere di basso valore, per avere il guadagno desiderato. In breve Rg, C devono soddisfare esigenze incompatibili: occorre pensare a qualcosa che permetta di avere una costante di tempo alta con Rg, C di piccolo valore. La soluzione, reperita su Internet, e da me adattata a questo circuito è in figura 4 dove i terminali w, z si collegano ad Rg1 ed Rg2, al posto del gruppo Rg – C in serie.

L'operazionale dovrebbe essere di basso offset, ho però usato un economico LF351.

Primo passo: In pratica l'operazionale, mantenendo una ddp prossima a zero ai suoi ingressi effettua una sorta di parallelo fra RG ed R *perché esse hanno praticamente la stessa tensione ai capi*. Essendo RG dell'ordine del megaohm, R delle decine di ohm di ha che RG//R = circa R e questo basso valore determinerà il guadagno di centro banda dell'AD524.

Secondo passo: la corrente che attraversa la capacità C è determinata da RG, in quanto l'ingresso positivo dell'operazionale LF351 è ad altissima impedenza. In sintesi la costante di tempo è RG * C, ma il guadagno è determinato da R! Il comportamento è così duale che ho voluto simularlo con SPICE; si può così vedere il diagramma di Bode (grafico della funzione di trasferimento,ampiezza asse x, frequenza asse

Elenco componenti AMPLIFICATORE	
R1	47 Ω 1/4 W
R2, R3	100 KΩ 1/2 W
R4	1 KΩ 1/2 W
R5	39 KΩ 1/2 W
R6, R7	10 KΩ 1/2 W
R8	1 MΩ 1/2 W
R9	Trimmer 100 KΩ cermet
C1	1 µF poliestere
C2, C6	47 µF 16V elettrolitico
C3	100 nF AVX
C4	22 µF 16V elettrolitico
C5	220 µF 16V elettrolitico
C7, C8	10 nF AVX
IC1	AD524
IC2	LF353
IC3, IC4	LF351

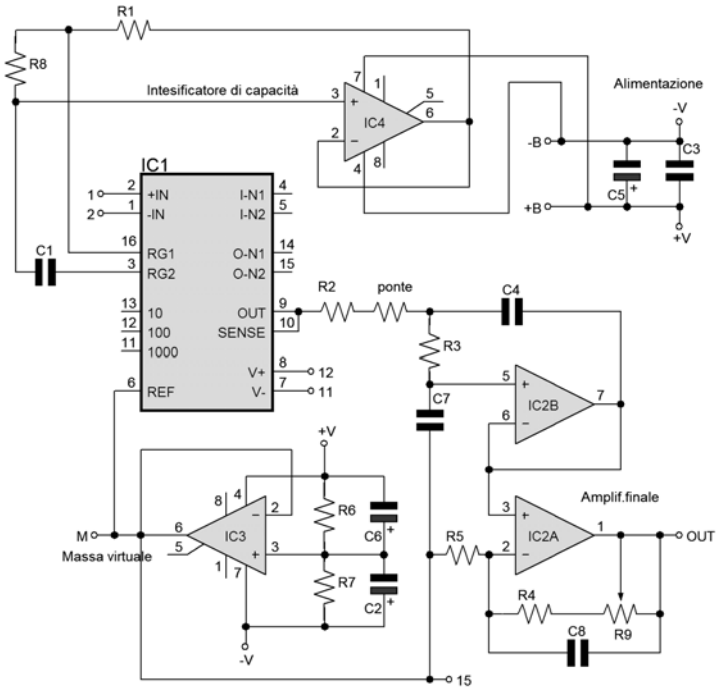


Figura 8 Schema elettrico dell'amplificatore ECG

25° MARC

mostramercato attrezzature radioamatoriali & componentistica
hardware • software
ricezione satellitare
editoria specializzata
radio d'epoca

Fiera di Genova
17-18 Dicembre 2005

sabato ore 9 • 18,30
domenica ore 9 • 18

ENTE PATROCINATORE:
A.R.I. - Ass. Radioamatori Italiani
Sezione di Genova
Salita Carbonara 65 b - 16125 Genova
C. P. 1117 - 16121 Genova - Tel./Fax 010.25.51.58
www.arigenova.it

ENTE ORGANIZZATORE E SEGRETERIA:
STUDIO FULCRO s.a.s.
Piazzale Kennedy, 1 - 16129 Genova
Tel. 010.561111 - Fax 010.590889
www.studio-fulcro.it e-mail: info@studio-fulcro.it

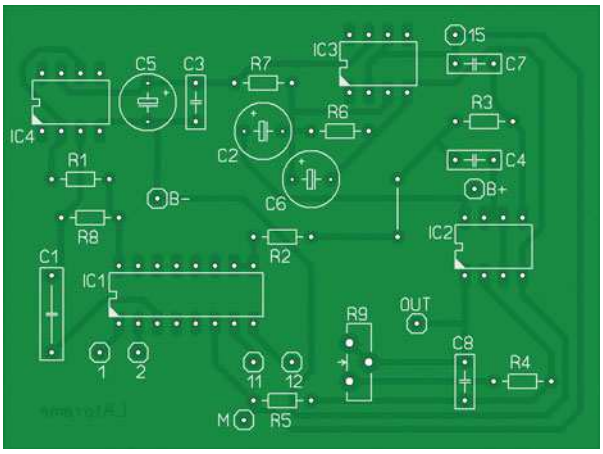


Figura 9 Piano di montaggio dell'amplificatore ECG

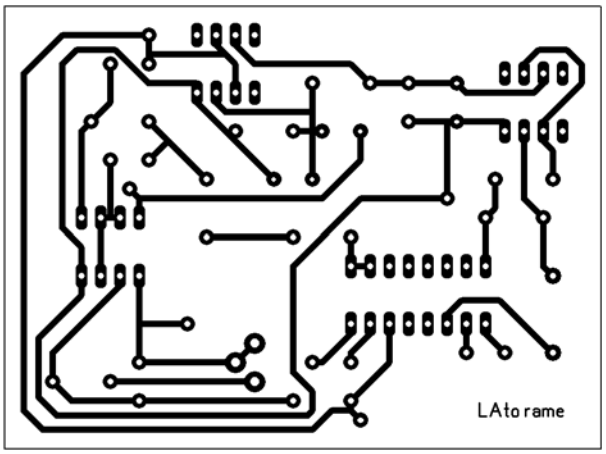


Figura 10 Circuito stampato dell'amplificatore EGC (lato rame – scala 1:1)

y) di un modello completo di filtro PA con moltiplicatore di capacità figura 5. la sua funzione di trasferimento è in figura 6 mentre nella 10 osserviamo la f.d.t. quando fra i terminali W - Z viene connessa una semplice serie R=75Ω C=1000µF: la differenza è notevole! Il taglio ad alta frequenza è sempre sui 140 Hz.

Per aumentare il rapporto S/N anticamente si usava collegare il paziente e anche il lettino a massa, tale pratica, foriera di incidenti mortali è stata abbandonata.
Nel nostro caso il corpo del soggetto viene connesso alla massa virtuale, ed il tutto è alimentato a batterie. Il nostro circuito è quindi costituito dall'intensificatore di capacità, che forma un PA per l'operazionale da strumentazione AD524, da un generatore di massa virtuale realizzato con LF351, da un filtro PB di *Chebyshev* a due poli ed un amplificatore finale, costruiti intorno ad un LF353, il semplice schema è in figura 8. Ricordarsi di collegare con due fili le alimentazioni di IC1 dai piedini 7 e 8 ai capi di C3!

IL CIRCUITO DI MODULAZIONE E TRASMISSIONE

Per poter trasmettere un segnale ECG, che ha una banda piuttosto bassa, la risposta raccomandata infatti dalle normative è 0.05 – 150 Hz, occorre renderlo compatibile con quanto richiesto in ingresso dal trasmettitore, che nel nostro caso è un TX FM audio *Aurel*, o simile realizzato con tecnologia SMD ed operante a 433.92 Mhz in modulazione di frequenza.
Tale TX ha queste specifiche: banda passante 20 Hz – 30 KHz, alimentazione da 9V fino 12 V sensibilità BF 100 mV pin 4, potenza RF 10mW su 50 Ohm.
Con il segnale ECG si pilota un modulatore, costruito intorno ad un CD4046 CMOS che fornirà il segnale di ingesso al TX. Il complesso è stato realizzato su una basetta separata, potendo così essere utilizzato per applicazioni più generali, la tensione massima di ingresso al modulatore è dell'ordine di 300 mV, la banda passante varia da 0.05 Hz a circa 200 hz con una frequenza di taglio di circa 100Hz ed una pendenza di 3 db per ottava.
Come funziona: l'integrato CD4046 è un PLL, Phased Locked Loop, un circuito ad aggancio di fase, contiene al suo interno un VCO, cioè un oscillatore RC controllato in tensione ed altra circuiteria che servirà nello stadio ricevente, in quanto useremo lo stesso compo-

La piedinatura del TX

1	+ 12V	9	Massa
2	Switch alimentazione (collegare al +12V)	10	NC
3	Massa	11	NC
4	Ingresso preamplificatore BF	12	NC
5	Massa	13	Massa
6	Uscita preamplificatore BF	14	NC
7	Entrata BF dal modulatore FM	15	Antenna 15 cm
8	NC	16	Massa

nente come demodulatore. Nel trasmettitore utilizziamo solamente il VCO che fornirà una frequenza variabile in uscita (pin 4) direttamente e linearmente proporzionale alla tensione applicata all'ingresso (pin 9) in pratica realizziamo un modulatore FM.
In assenza di pilotaggio la frequenza a riposo del VCO viene impostata tramite il trimmer R4 la capacità di ingresso C4 e la resistenza R5 sono scelte in modo da avere una buona risposta anche quando il segnale modulante ha componenti a frazioni di Hertz.
Per avere una buona banda passante di tutto il complesso è bene scegliere la frequenza a riposo del VCO più alta possibile compatibilmente con la coppia di TX, RX utilizzati, questa si è rivelata essere circa 7Khz e dipende da R1, C1. mentre fmax/fmin di aggancio è funzione di R2/R1.

Elenco componenti Modulatore TX	
R1, R5	12 KΩ 1/4 W
R2	120 KΩ 1/4 W
R3	Trimmer 10 KΩ cermet
R4	Trimmer 20 KΩ cermet
R6	47 Ω 1/4 W
C1	10 nF AVX
C2	4.7 µF 16V elettrolitico
C3	470 µF 16V elettrolitico
C4,C5	220 µF 16V elettrolitico
TX	TX Aurel FM audio

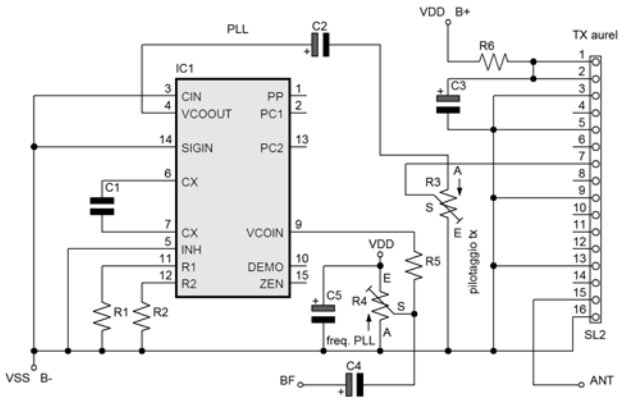


Figura 11 Schema elettrico del Modulatore TX

27a MOSTRA REGIONALE DELL'ELETTRONICA
18-19 FEBBRAIO 2006
SCANDIANO (RE)

componentistica / computer / hi-fi car
telefonia / radiantismo CB e OM
videoregistrazione
mercatino delle pulci radioamatoriali
CENTRO FIERISTICO SCANDIANO
Tel. 0522.857436 www.fierascandiano.it
orari: sabato 18 dalle 9.00 alle 18.30
domenica 19 dalle 9.00 alle 18.00

Il trimmer R4 regola il segnale in ingresso al modulatore e va tarato osservando il segnale ricevuto. Il trimmer R3 dosa il segnale in ingresso al TX.

IL CIRCUITO DI RICEZIONE

Il ricevitore è un RX Aurel FM audio 433.92 Mhz., la cui uscita BF è dell'ordine di 30mV.
Specifiche: banda passante come TX, alimentazione 3V, uscita BF 90 mVeff ad 1 KHz massima modulazione, sensibilità 100 dBm.

La piedinatura del RX

1	+3V	11	Massa
2	Massa	12	NC
3	Antenna	13	NC
4	NC	14	NC
5	NC	15	Comando dello squelch
6	NC	16	Massa
7	Massa	17	NC
8	NC	18	Uscita BF con squelch
9	NC	19	Interruttore comandato dallo squelch
10	Uscita BF	20	Massa

Occorre portare il segnale ricevuto a livello utile: a tale scopo il transistor Q1 effettua una prima amplificazione fino a 3Vpp, in seguito dopo aver effettuato una opportuna traslazione di componente continua tramite il partitore a resistenze eguali R7, R6, due sezioni in serie del trigger CD4093 eliminano qualunque componente a modulazione di ampiezza fornendo una uscita pulita e compatibile col demodulatore, come da schema elettrico in figura 16.
Come già accennato usiamo il CD4046, però configurato nella ricevente come anello di reazione ad aggancio di fase. In pratica la frequenza di oscillazione del VCO, frequenza interna (Fint),viene comandata da una tensione proporzionale a (Fint-Fing) dove Fing è applicata al pin 14 ed è una frequenza di ingresso i cui valori massimi e minimi sono compatibili con le RC tramite le quali il VCO oscilla.
In altre parole, se si applica al piedino 14 una Fing compatibile con il campo di aggancio del PLL viene generata per mezzo dell'OR esclusivo XOR ed il gruppo di integrazione R9 C9 una tensione che retroazionata al pin 9 agisce in modo da rendere minima la differenza Fint-Fing. Se Fing varia nel tempo, e nel nostro caso varia seguendo il segnale ECG, la tensione di retroazione, che si ritrova anche sul pin 10 segue fedelmente tale variazione.
Quindi sul pin 10 avremo il desiderato segnale, inquinato però da un ripple di conversione, ed altri disturbi e sarà necessario usare un filtro PB costituito da R10, C10.
Il filtro PB - R9, C9 – pur essendo del primo ordine, deve essere attentamente dimensionato, in quanto fa parte di un loop di reazione che si comporta complessivamente come un sistema di 2° grado con i soliti problemi legati a ζ.
Ho trovato sperimentalmente R9 utilizzan-

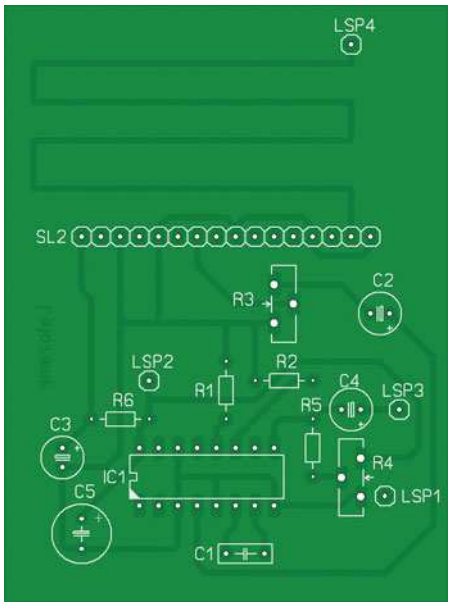


Figura 12 Piano di montaggio del Modulatore TX

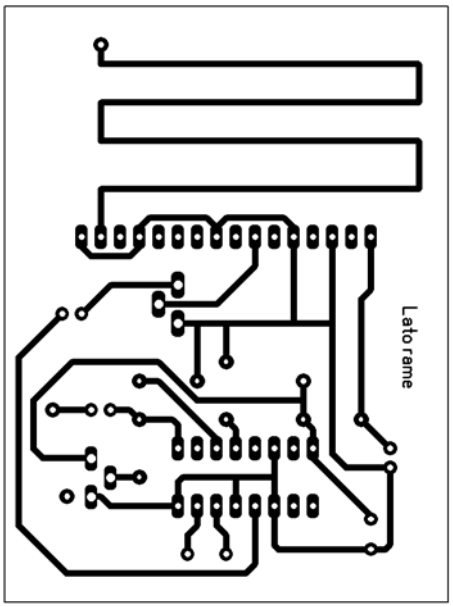


Figura 13 Circuito stampato del Modulatore TX (lato rame – scala 1:1)

In Edicola sino a fine Gennaio

Da non Perdere!

Contiene cartolina per abbonarsi a Fare Elettronica a condizioni vantaggiose

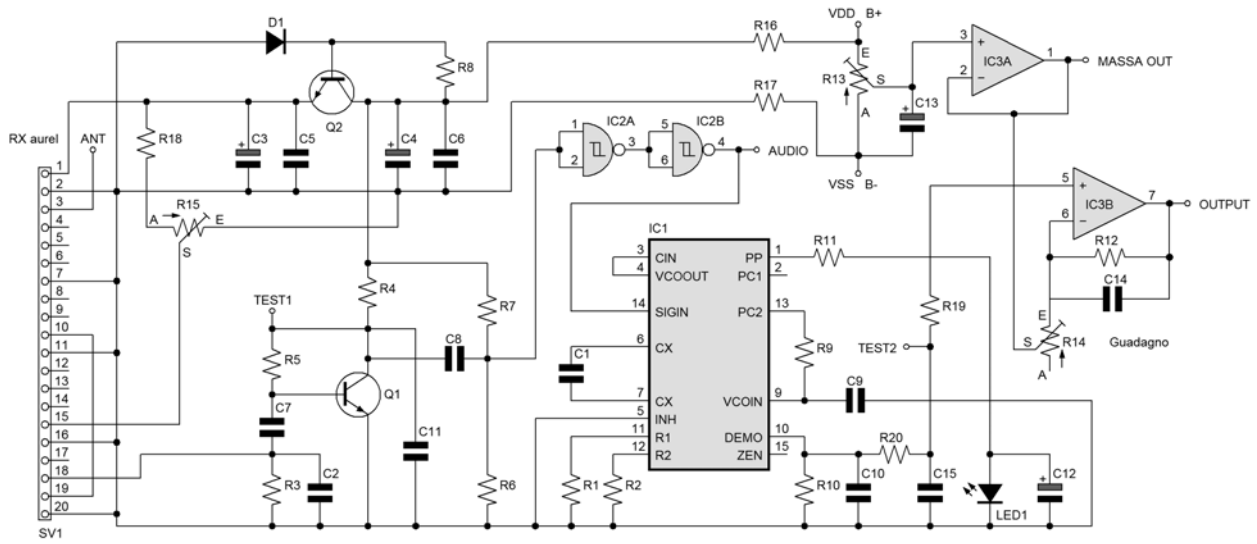


Figura 14 Schema elettrico del Demodulatore RX

do un trimmer. effettuando una taratura finché la risposta all'oscilloscopio non mi è sembrata soddisfacente. Il transistor Q2 è collegato come alimentatore e provvede i 3V necessari all'alimentazione dello RX. IC3a fornisce una tensione continua regolabile a piacere, che rappresenta il riferimento di massa all'uscita.
Regolando R13 si centra il tracciato sullo schermo dell'oscilloscopio, regolando R14 si aggiusta il guadagno: IC3b è configurato come amplificatore non invertente. Il led collegato al pin 1 si illumina quando c'è aggancio di fase tra i PLL dell'unità trasmittente e ricevente.

MESSA A PUNTO E TARATURA

Amplificatore

Mettere a metà corsa il trimmer R9, cortocircuitare tra loro i tre ingressi, massa virtuale-pad1-pad2, alimentare con 9V, l'assorbimento dovrebbe aggirarsi sui 10–13 mA.
Controllare con oscilloscopio l'assenza di auto-oscillazioni, in loro presenza saldare, direttamente sul retro dello stampato, terminali corti, condensatori AVX da 100 nF tra pin 8, pin 6 integrato IC1, eventualmente pin 6, pin 4 di IC3, e nei casi più riottosi anche sui pin di alimentazione di IC4, IC2. Il guadagno è circa 1000 e alcuni miei prototipi oscillavano, altri no.
In seguito togliere il corto sugli ingressi, col-

Elenco componenti Demodulatore RX	
R1, R9	12 KΩ 1/4 W
R2, R12	120 KΩ 1/4 W
R3	Vedi testo
R4	5,6 KΩ 1/4 W
R5	1 MΩ 1/4 W
R6, R7	56 KΩ 1/4 W
R8	1 KΩ 1/4 W
R10	100 KΩ 1/4 W
R11	150 Ω 1/4 W
R13	Trimmer 20 KΩ cermet
R14	Trimmer 50 KΩ cermet
R15	Trimmer 10 KΩ cermet
R16, R17	Ponticelli
R18	10 KΩ 1/4 W
R19, R20	47 KΩ 1/4 W
C1, C11	10 nF AVX
C2	Vedi testo
C3, C13	100 µF 16V elettrolitico
C4	220 µF 16V elettrolitico
C5÷C8, C10	220 nF AVX
C9	47 nF AVX
C12	2.2 µF 16V elettrolitico
C14, C15	22 nF AVX
D1	Zener 3.9V 1/2 W
LED1	Led alta luminosità
Q1, Q2	BC547
IC1	CD4046
IC2	CD4093
IC3	LF353
RX	RX Aurel FM audio

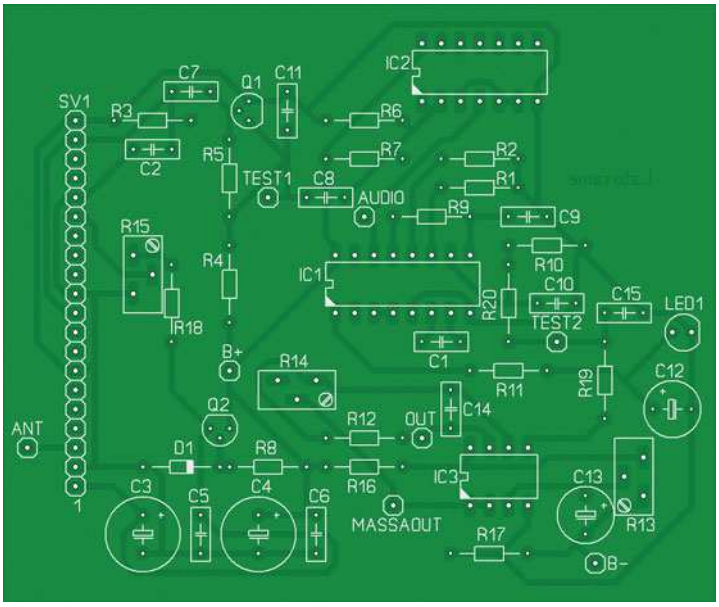


Figura 15 Piano di montaggio del Demodulatore RX

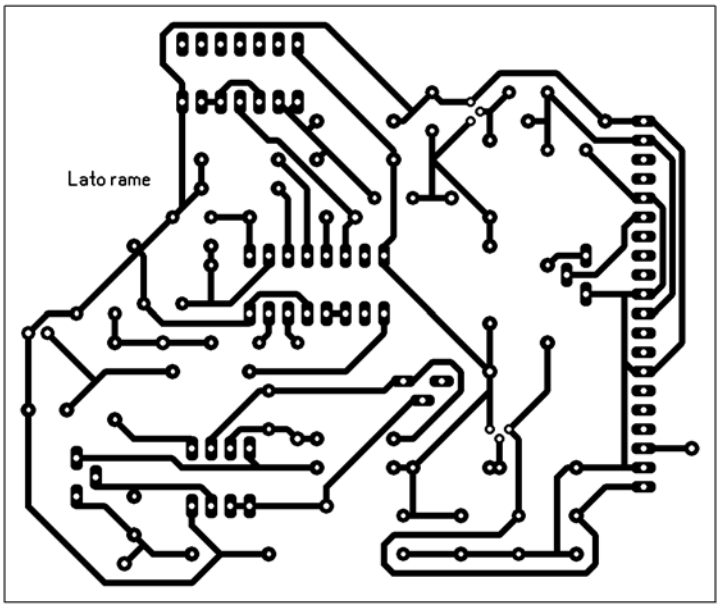


Figura 16 Circuito stampato del Demodulatore RX (lato rame – scala 1:1)

legare il SIMULATORE, come descritto nella *prima parte*: si dovrebbe osservare un segnale come in figura 18.

Gruppo RX –TX

La ricevente accetta come massima alimentazione 3.3V, è connessa allo stampato tramite una striscia di 20 pin a tulipano spaziatura 1/10". È prudente prima controllare la tensione ai capi di C3 e poi inserirla. Mettere tutti i trimmer a metà corsa, alimentare con 9V. Collegare l'oscilloscopio tra MASSAOUT e OUTPL, provare il funzionamento dello squelch, trimmer R15, in assenza di segnale TX si dovrebbero osservare segnali casuali, che agganciano occasionalmente il PLL e accendono il led. Regolare il trimmer un poco oltre il silenziamento. Accendere ora il TX, sempre a 9V, trimmer a metà corsa, si osserva l'aggancio del PLL ed un nitido segnale rettangolare sull'uscita AUDIO.
Modulando ora il TX con un segnale BF, circa 300 mVpp di frequenza 30 Hz regolare i trimmer PILOTAGGIO, FREQUENZA PLL per il miglior segnale ricevuto su TEST2. La portata del complesso è circa 30m tra pareti di cemento armato. I componenti R3, C3 che avevo all'inizio messo per fornire un certo carico alla ricevente si sono rivelati superflui, il trimmer R13 controlla l'offset. R14 il guadagno. Finalmente collegare SIMULATORE, amplificatore, TX come in figura 19

Il risultato finale si può vedere in figura 20 dove sono rappresentate la traccia trasmessa, superiore e quella ricevuta, inferiore. Asse Y 1V/div,asse X 50 mSec/div.
Regolare opportunamente l'offset ed il guadagno sia tale da avere circa 1V in uscita dalla ricevente.

Collegamento finale

Alimentare il ricevitore, collegarlo all'oscilloscopio, regolare sensibilità e base dei tempi. Scollegare il SIMULATORE e connettere ai tre ingressi gli elettrodi ed i medesimi al corpo, come spiegato nella prima parte, alimentazione **rigorosamente a pile**! Il TX inizia a trasmettere e i due PLL si agganciano, si accende il LED della ricevente.
Dopo *una trentina di secondi* apparirà sullo schermo il segnale cardiaco. Una ulteriore riduzione del rumore a 50Hz si può ottenere usando cavetti schermati per connettere gli elettrodi all'amplificatore. Il collegamento di massa alla calza schermante *deve essere effettuato* solamente dal lato del front end.

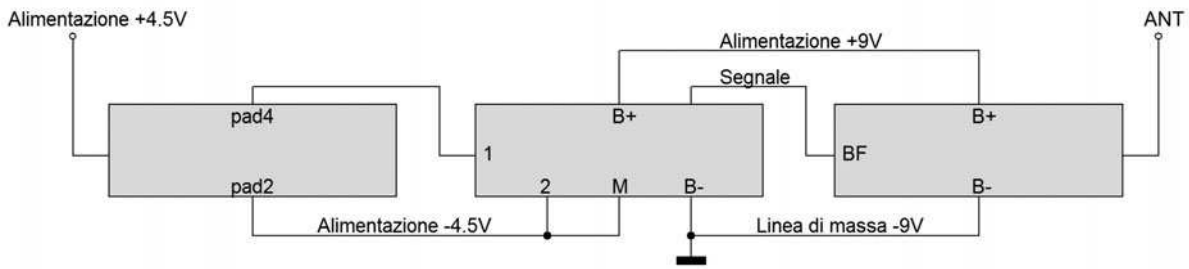


Figura 17 Schema a blocchi complesso trasmittente

Note di costruzione

L'assorbimento dell'amplificatore è circa 12 mA, quello del TX + modulatore circa 16 mA, è bene provvedere all'alimentazione usando due pile piatte da 4,5 V in serie, Conviene inscatolare l'amplificatore in un piccolo contenitore metallico che sarà poi contenuto da uno di plastica, dove troveranno alloggio le pile ed il gruppo Modulatore TX FM.
Per comodità, ho realizzato l'antenna del TX direttamente sullo stampato, vista però l'alta frequenza in gioco occorrerebbe una basetta a minima perdita: nulla vieta di realizzarla con uno spezzone di filo di circa 15 cm. La frequenza di oscillazione del CD4046 è estremamente sensibile alla tensione di alimentazione, se si prevede un lungo uso del complesso si possono stabilizzare le alimentazioni dei due PLL utilizzando lo zener interno che fa capo al pin 15. Il ricevitore-demodulatore può essere alloggiato in un contenitore plastico, mettendo in evidenza il led del PLL che rivelerà l'aggancio di fase. Attenzione: la massa a cui si collega l'oscilloscopio non è quella di alimentazione!
È possibile ascoltare il segnale modulato prelevandolo dall'uscita BF, oppure utilizzando qualunque RX scanner.

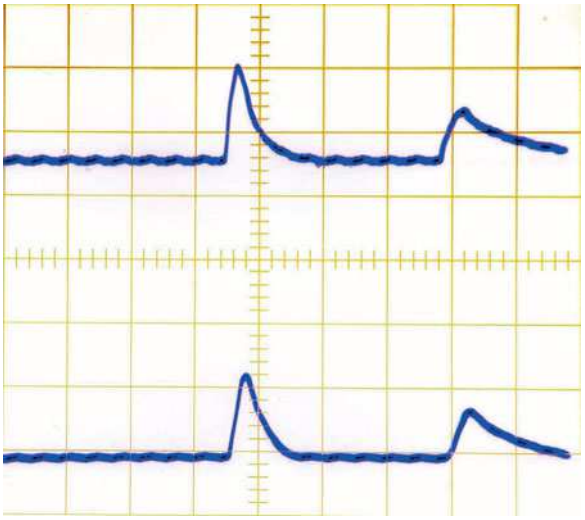


Figura 18 Tracce oscilloscopiche del segnale trasmesso e ricevuto

ALCUNI INTERESSANTI SVILUPPI

PRIMO: Verso un sistema di analisi automatica: nel corso del tempo sono andati sviluppandosi sistemi di analisi automatica basati sullo sviluppo di algoritmi di classificazione che acquisiscono i dati di imput dal campionamento del segnale, dalla frequenza del medesimo alla componente continua all'area sottesa dagli impulsi positivi e negativi ecc. Ancora più recentemente altre informazioni possono essere acquisite analizzando il contenuto armonico, ad esempio tramite FFT. Tali tecniche sono oggetto anche di esercitazioni di laboratorio presso alcune università. Esse hanno in comune una conoscenza esplicita degli algoritmi di ricerca e classificazione. Interessante ed innovativo è l'uso, per un sistema di diagnosi automatica, di Reti Neurali implementate tramite hardware o software, quali Visual Basic o il gratuito Octave che è praticamente un clone di Matlab, un potente linguaggio di programmazione matematica. Questa tecnica, che potrebbe sembrare esclusivo appannaggio di costosi Enti di ricerca, è fattibile anche, con un minimo di studio, a livello amatoriale. Occorre implementare un generatore di segnale ECG che simuli a richiesta tracciati sani o affetti da patologie e ciò si può fare mappando un campionamento numerico dei tracciati nella memoria di un microprocessore che restituisce una uscita in tensione tramite una scansione temporale dei dati imma-

CITTÀ
DI MONTEROTONDO

11 – 12 FEBBRAIO 2006

18ª FIERA
DEL RADIOAMATORE
DELL'ELETTRONICA
ED INFORMATICA

MONTEROTONDO (ROMA)

PALAZZETTO DELLO SPORT

ORE 9.00 – 13.00 / 15.00 – 19.00

ORGANIZZAZIONE GEDIT

TEL. 075.5991028 – 348.3307007

gazzinati ed una conversione digitale / analogica, seguita da un filtro passa basso. In seguito i segnali vengono acquisiti da una rete neurale, tramite un procedimento di campionamento. Forzando le corrispondenze fra tracciati e patologie associate, la rete apprende in modo implicito,quindi non trasparente all'utente, ed effettua le classificazioni volute: essa esegue anche, se ben progettata, le interpolazioni fra patologie. Il critico numero di unità di ingresso, nascoste e di uscita si può ottenere tramite algoritmi evolutivi genetici.

SECONDO: Una ricerca sui punti dell'agopuntura, argomento meno mistico di quanto potrebbe sembrare, in quanto frutto di pratica millenaria e comunque oggetto al giorno d'oggi di un mercato nel quale girano fior di soldoni. In pratica,la stimolazione di opportune aree del corpo, di area molto piccola, situate lungo particolari percorsi, chiamati meridiani, permette,secondo tale scuola, di esercitare azione terapeutica. La stimolazione può essere effettuata mediante aghi di diverso tipo, tensioni elettriche, calore (Moxa), recentemente anche tramite laser o luce colorata puntiforme – con diversi colori. Disponendo di una sonda elettrica quasi puntiforme, con un CMRR tale da poter agire a secco,senza gel conduttivo, si potrebbe indagare se ad una stimolazione non elettrica in un punto corrisponda una alterazione dell'attività elettrica in un altro punto che abbia corrispondenza terapeutica con quello stimolato. I punti hanno area piccola, la pasta elettroconduttiva medierebbe inevitabilmente il loro potenziale su una superficie più vasta e pertanto per compensare il degradamento del segnale acquisito occorre una grande reiezione di modo comune. La stimolazione non deve essere di natura elettrica, per non mascherare il segnale eventualmente ottenuto. Per quanto ne so è una ricerca aperta e se qualcuno è interessato può contattarmi.

TERZO: Avendo a disposizione un modulatore/demodulatore FM possiamo registrare l'ECG su una normale cassetta audio. Questa tecnica è una sorta di Horner, procedimento anni 60 mediante la quale si effettuava una registrazione del ritmo cardiaco nell'arco di una giornata, con uno strumento portatile. In seguito il nastro veniva riprodotto a velocità accelerata e si ricercavano eventuali anomalie, ad es. in frequenza. Una evoluzione di questo procedimento consiste nel campionare il segnale a qualche migliaio di Hz, effettuare una conversione A/D a 12 o 16 bit, memorizzarlo su una ram, ad esempio una Secure Digital SD. Dalla SD i dati possono essere estratti a grande velocità ed analizzati con tecniche di riconoscimento digitali.

CONCLUSIONI

Abbiamo fino a qui visto come costruire il primo sensore del nostro sistema di biofeedback, il rivelatore di segnale cardiaco. Qualcosa di semplice da costruire, poco costosa, che può rivaleggiare con ben più costosi apparati commerciali. Al momento potremo usare questo strumento per osservare il nostro tracciato cardiaco a riposo oppure in movimento,usando però gli opportuni elettrodi adesivi, mentre ad esempio facciamo bicicletta da camera ecc. Nelle prossime puntate vedremo come funzionano e come costruire altri sensori, alcuni dei quali sono venduti a prezzi stratosferici. Nella parte finale di questo ciclo dedicato al biofeedbak verrà mostrato come elaborare i segnali acquisiti, come registrarli, come presentarli in forma utile e sintetica al soggetto dal quale vengono prelevati.

AVVERTENZA IMPORTANTE

Questo strumento è proposto a scopo didattico sperimentale, non ha alcun tipo di certificazione. NON collegare la parte trasmittente a qualunque apparecchiatura connessa alla rete luce, come oscilloscopi, anche del tipo a batteria se con tubo CRT, alimentatori, eccetera, quando gli elettrodi sensori (anche solo uno) sono a contatto del corpo. Si ricorda che piccolissime correnti circolanti nell'organismo possono condurre ad esiti fatali e che la resistenza di contatto tra elettrodi e pelle può diventare molto bassa.

fare elettronica

CAMPAGNA ABBONAMENTI 2005/ 2006

1 ANNO, 11 RIVISTE

A SOLI 45,00 EURO INVECE DI 60,50 EURO

con un risparmio del

25%

Abbonati subito!

Compila oggi stesso il coupon che trovi in ultima pagina o abbonati su www.farelettronica.com/abbonamento

Ottava parte
n° 245 - Novembre 2005
L'amplificatore differenziale

Nona parte
n° 246 - Dicembre 2005
L'operazionale a reazione di corrente

Decima parte
n° 247 - Gennaio 2006
L'alimentazione degli op-amp

L'amplificatore operazionale dalla A alla Z

L'operazionale a reazione di corrente è il primo op-amp che ha consentito l'estensione dell'utilizzazione del componente in alta frequenza.

GENERALITÀ

Più recentemente sono stati introdotti nel mercato della componentistica attiva nuovi amplificatori operazionali detti a reazione di corrente, o CFA (*Current Feedback Amplifier*), la cui peculiare caratteristica risiede in un'elevata banda passante che, indipendente dal guadagno entro ampi limiti, consente frequenze elevate del tutto impensabili fino a pochi anni fa.
L'impiego dei CFA è praticamente illimitato potendo andare a sostituire nella maggior parte dei casi, gli operazionali a reazione di tensione che, almeno fin qui, rimangono ancora favoriti da parte dei progettisti.
Questa preferenza è dovuta anche al fatto che, in tempi relativamente recenti, le Case costruttrici hanno prodotto VFA (*Voltage Feedback Amplifier*) con prodotto guadagno per banda passante notevolmente elevato. A questo proposito basti pensare al classico, quasi storico, LM741 che presenta un *GBW* di 1 MHz e, per esempio, al CLC420 della National o all'OPA2652 della Burr-Brown o al 4350 della Maxim che presentano una banda passante che va oltre i 200 MHz; per non citare modelli che ormai approssimano il *GBW* al GHz.
Le caratteristiche dei CFA e le modalità progettuali dei relativi circuiti possono meglio evidenziarsi ponendoli a confronto con i VFA. Gli operazionali a reazione di tensione, i VFA, presentano:

- un più basso rumore;
- un migliore comportamento rispetto alle componenti continue;
- minime limitazioni nella realizzazione del canale di reazione;
- una notevole contrazione della banda passante all'aumentare del guadagno.

Per contro, gli operazionali a reazione di corrente, i CFA, presentano:

- un più elevato slewrate;
- una più bassa distorsione;
- limitazioni (ma ininfluenti) sull'anello di reazione;
- una banda passante più ampia e pressoché costante al variare del guadagno.

La figura 9.1 *a* riporta il noto modello a cui può essere ricondotto l'amplificatore operazionale a reazione di tensione ad anello aperto, mentre la figura 9.1 *b* il riporta lo stesso modello ma reazionato, ossia ad anello chiuso.

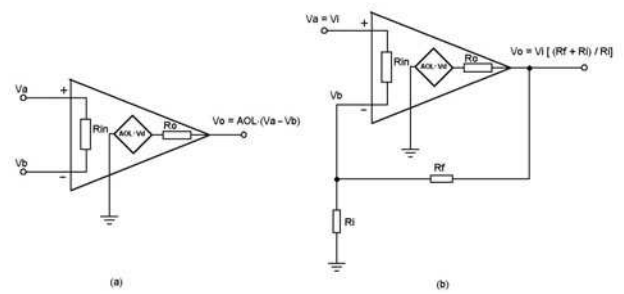


Figura 9.1 *Modello dell'op-amp a reazione di tensione ad anello aperto in (a) e ad anello chiuso in (b)*

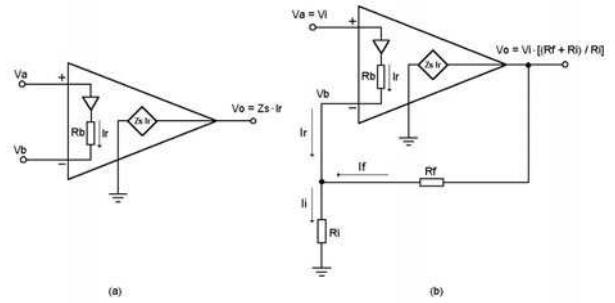
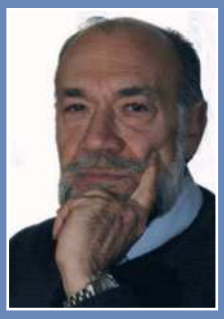


Figura 9.2 *Modello dell'op-amp a reazione di corrente ad anello aperto in (a) e ad anello chiuso in (b)*

L'operazionale a reazione di corrente



di Nico Grilloni
(n.grilloni@farelettronica.com)

9.2 L'OPERAZIONALE A REAZIONE DI CORRENTE. IL GUADAGNO

Si consideri pertanto il modello dell'amplificatore operazionale a reazione di tensione riportato nella figura 9.1 *a*. Nel caso ideale, come si è già detto, entrambi gli ingressi hanno impedenza infinita, mentre il circuito di uscita è assimilabile a un generatore controllato di tensione $A_{OL}\ V_d = A_{OL}\ (V_a - V_b)$, con resistenza R_o prossima a zero, e dipendente dalla frequenza nel senso che il guadagno diminuisce all'aumentare della frequenza.

La differenza di potenziale $(V_a - V_b)$ che si manifesta fra i due ingressi controlla il generatore di tensione, mentre il circuito di reazione - figura 9.1 *b* - forza a zero la differenza $(V_a - V_b)$ tendendo a portare le tensioni sui due ingressi allo stesso valore (da qui la definizione di op-amp a reazione di tensione). Come si è già esposto negli articoli precedenti, infatti, i due ingressi di un amplificatore reazionato tendono ad assumere lo stesso potenziale. Per il guadagno reale dell'operazionale reazionato in configurazione non invertente, si è giunti all'espressione [9.7] In questa si è evidenziato come l'espressione reale del guadagno si discosti dal valore ideale per un termine, definito di errore, che però, con l'approssimazione $A_{OL} \to \infty$, può essere trascurato. La [9.7]è qui riproposta nella [9.1]per comodità.

$$A_{CL} = \frac{R_i + R_f}{R_i\left[\dfrac{R_f + R_i}{R_i \cdot A_{OL}} + 1\right]} = \frac{R_i + R_f}{R_i} \cdot \frac{1}{1 + \dfrac{1}{A_{OL}\ \beta}} \qquad [9.1]$$

Tenendo presente che il rapporto $[\ R_i + R_f)/R_i]$ indica il guadagno ideale A_{CLi}, e che $\beta = 1/A_{CLi}$, la [9.1] come si è visto, può porsi nella forma:

$$A_{CL} = \frac{V_o}{V_i} = \frac{A_{CLi}}{1 + \dfrac{A_{CLi}}{A_{OL}}} \qquad [9.2]$$

Si consideri adesso quanto illustrato nella figura 9.2 *a* dove è riportato il modello a cui è riconducibile l'op-amp a reazione di corrente.

Fra gli ingressi invertente e non invertente è posto un buffer a guadagno unitario. La presenza del buffer fa sì che l'impedenza di ingresso dell'input non invertente sia teoricamente infinita, mentre sia prossima a zero l'impedenza dell'input invertente (coincidente con l'uscita del buffer). Questo è un primo elemento che differenzia i CFA dai VFA. Questi ultimi infatti, presentano impedenza infinita su entrambi gli ingressi. La funzione del buffer, oltre a consentire alla tensione sull'ingresso invertente di seguire le escursioni del segnale presente sull'input non invertente, fornisce un percorso a bassa impedenza alla corrente I_r, detta corrente di errore. Lo stadio di uscita del CFA è un generatore di tensione con resistenza di uscita nulla controllato dalla corrente I_r. La presenza della reazione forza a zero questa corrente. Da qui la definizione di operazionale a reazione di corrente. L'amplificazione è fornita dalla transimpedenza Z_s (guadagno di transimpedenza che è funzione della frequenza) che, risentendo le variazioni della corrente I_r, genera un segnale di uscita $V_o = Z_s \cdot I_r$. Per ricavare la funzione di trasferimento del CFA è sufficiente considerare che, con riferimento alla figura 9.2 *b*, si ha:

$$I_r = I_i - I_f$$

Ossia, essendo $V_b = V_a = V_i$:

$$I_r = (V_i/R_i) - [\ V_o - V_i)\ /R_f] = (\ V_i/R_i) + [\ V_i - V_o)\ /R_f]$$

Ma essendo anche $I_r = V_o\ /Z_s$ si ricava:

$$V_o\ /Z_s = (V_i/R_i) + [\ V_i - V_o)\ /R_f]$$

Da questa, per il rapporto $V_o/V_i = A_{CL}$, si ottiene:

$$A_{CL} = \frac{V_o}{V_i} = \frac{\dfrac{1}{R_f} + \dfrac{1}{R_i}}{\dfrac{1}{Z_s} + \dfrac{1}{R_f}}$$

da cui, posto il rapporto $(R_f + R_i) / R_i = A_{CLi}$, si ricava:

$$A_{CL} = \frac{V_o}{V_i} = \frac{A_{CLi}}{1 + \frac{R_f}{Z_s}} = \frac{A_{CLi}}{1 + \frac{1}{K'}} \qquad [9.3]$$

dove $K' = Z_s/R_f$ è il guadagno di anello.
Per $R_f = Z_s$ si ha un dimezzamento del guadagno, mentre per $Z_s \to \infty$, ovvero per $I_r \to 0$, il rapporto V_o/V_i tende ad A_{CLi} [$A_{CLi} = 1 + (R_f/R_i)$]
Da ciò si deduce che per avere un guadagno prossimo al valore ideale è necessario che l'op-amp abbia un valore di Z_s quanto più possibile elevato. Per il CFA in configurazione invertente valgono le stesse considerazioni. È da tenere presente che, in questo caso, il guadagno è espresso dal rapporto R_f/R_i.

9.3 RISPOSTA IN FREQUENZA DEI CFA (R_b = 0)

Il comportamento della transimpedenza in funzione della frequenza, e ad anello aperto – figura 9.3 *b* – è rappresentato da curve il cui andamento è simile all'andamento delle curve del guadagno $A_{OL}(f)$ – figura 9.3 *a* – già visto per gli operazionali a reazione di tensione. Si nota infatti, un primo tratto costante fino ad una certa frequenza aldilà della quale il guadagno di transimpedenza decresce con un'attenuazione di 20 dB per decade.
Ciò è dovuto alla struttura interna dell'IC che crea delle capacità assimilabili ad un'unica capacità il cui valore è dell'ordine del decimo di pF. Questa capacità realizza comunque un filtro passa-basso del primo ordine la cui attenuazione è appunto di 20 dB/decade.
Indicando con Z_o il valore della transimpedenza Z_s per $f = 0$, ossia in continua, e con f_t il valore della frequenza a –3 dB, si può ricavare l'andamento di Z_s in funzione di f con un'espressione del tutto simile a quella già

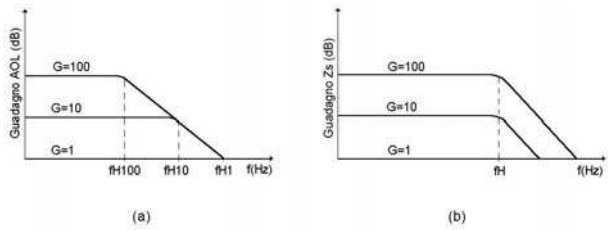


Figura 9.3 *A confronto le curve di risposta dell'operazionale a reazione di tensione, (a), e dell'operazionale a reazione di corrente, (b). Nei CFA la frequenza di taglio ha un valore notevolmente elevato e si mantiene costante, entro ampi limiti, al variare del guadagno*

ricavata per gli operazionali a reazione di tensione (cfr. [1.16]). Si ha infatti l'espressione complessa:

$$Z_s(f) = Z_o / [1 + j(f/f_t)] \qquad [9.4]$$

il cui modulo è:

$$Z_s(f) = \frac{Z_o}{\sqrt{1 + \left(\frac{f}{f_t}\right)^2}} \qquad [9.5]$$

Pertanto per $f = 0$ si ha $Z_s(f) = Z_o$, per $f = f_t$ si ha $Z_s(f) = 0{,}707\ Z_o$ (–3 dB), e, infine, per $f \to \infty$ si ha $Z_s(f) \to 0$. Sostituendo la [9.4] nella [9.3] e tenendo conto che $R_f/Z_o << 1$, si ricava:

$$A_{CL}(f) = \frac{A_{CLi}}{1 + j\frac{R_f \cdot f}{Z_o \cdot f_t}} \qquad [9.6]$$

Ponendo quindi:

$$f_H = (Z_o \cdot f_t) / R_f \qquad [9.7]$$

la [9.6] in modulo, assume l'espressione:

$$A_{CL}(f) = \frac{A_{CLi}}{\sqrt{1 + \left(\frac{f}{f_H}\right)^2}} \qquad [9.8]$$

La [9.7] rappresenta la frequenza di taglio superiore, ovvero la banda passante ad anello chiuso considerando eguale a zero la resistenza R_b di uscita del buffer, mentre la [9.8] esprime il guadagno ad anello chiuso in funzione della frequenza.
Confrontando le espressioni [9.2] e [9.3] si constata come siano fra loro molto simili le funzioni di trasferimento dei CFA e dei VFA.
Nella [9.3] per $Z_s \to \infty$ si ha infatti il caso ideale per il quale è $V_o/V_i = A_{CLi}$. Analogamente, nella [9.2] per $A_{OL} \to \infty$ si ha $A_{CL} = A_{CLi}$.
Confrontando poi la [9.5] con la [1.17] si vede come le due espressioni siano identiche ove al guadagno di transimpedenza del CFA si sostituisca il guadagno ad anello aperto del VFA.
La differenza sostanziale sta invece nel fatto che nel caso dei VFA il guadagno può essere impostato modificando sia il valore attribuito alla resistenza R_f che il valore attribuito alla resistenza R_i, mentre nel caso dei CFA alla resistenza R_f è necessario attribuire il valore sug-

gerito dal costruttore.
L'espressione [9.7] infatti, evidenzia la dipendenza della frequenza f_H dalla resistenza R_f. Pertanto a questa resistenza si assegnerà il compito di definire la banda passante, mentre alla resistenza R_i si affiderà il compito di modificare il guadagno secondo le esigenze.
La banda passante è, pertanto, indipendente dal guadagno ed è per questo motivo che per i CFA non viene specificato il classico parametro *GBW*, ma solo la banda passante.
Se, viceversa, per modificare il guadagno si agisce sul valore della resistenza R_f, si dovrà sempre tenere presente l'espressione [9.7] che indica come a un incremento del valore della medesima corrisponde una contrazione della banda passante (in genere duplicando la R_f si ha una riduzione del 50 % della banda passante).
In ogni caso è però sempre bene attenersi, per la R_f, ai valori forniti dalle note tecniche delle Case costruttrici in funzione del guadagno e della tensione di alimentazione adottata. Alterando il valore della R_f senza precise cognizioni si rischia l'instabilità dell'op-amp.

9.3.1 Risposta in frequenza dei CFA ($R_b \neq 0$)

Si consideri adesso il caso reale per il quale la resistenza di uscita R_b del buffer è diversa da zero. Con riferimento alla figura 9.2 *b*, per le correnti al nodo si ha:

$$I_r = I_i - I_f$$

ossia:

$$I_r = \frac{V_b}{R_i} - \frac{V_o - V_b}{R_f} \qquad [9.9]$$

Considerando il guadagno del buffer unitario, si ha per V_b:

$$V_b = V_a - R_b\, I_r \qquad [9.10]$$

Essendo $I_r = V_o / Z_s$, sostituendo quest'ultima e la [9.10] nella [9.9] si ricava:

$$\frac{V_o}{Z_s} \cdot R_f + V_o = \left(V_a - R_b \frac{V_o}{Z_s} \right) \cdot \left(1 + \frac{R_f}{R_i} \right) \qquad [9.11]$$

Risolvendo questa per V_o/V_a, ossia per V_o/V_i (coincidendo V_a con V_i), si ha:

$$\frac{V_o}{V_i} = \frac{1 + \frac{R_f}{R_i}}{1 + \frac{R_f + R_b \cdot \left(1 + \frac{R_f}{R_i} \right)}{Z_s}} \qquad [9.12]$$

In questa espressione il termine $[1 + (R_f/R_i)]$ è il guadagno ideale, mentre il termine:

$$\frac{1}{K'} = \frac{R_f + R_b \cdot \left(1 + \frac{R_f}{R_i} \right)}{Z_s} \qquad [9.13]$$

è il reciproco del guadagno di anello K' per R_b diversa da zero. Riprendendo l'espressione [9.7] e sempre indicando con Z_o la transimpedenza in continua (ossia per $f = 0$ Hz), tenendo conto della [9.13] si ha:

$$f_H = \frac{\frac{Z_o f_T}{R_f}}{1 + \frac{R_b}{R_f} \cdot \left(1 + \frac{R_f}{R_i} \right)} \qquad [9.14]$$

Confrontando quest'ultima con la [9.7] si vede che la presenza di una resistenza di uscita del buffer diversa da zero determina una contrazione della banda passante.
Questa contrazione sarà tanto più significativa quanto più elevato sarà il guadagno $[1 + (R_f/R_i)]$ attribuito allo stadio amplificatore.
Pertanto, nella definizione della banda passante di un amplificatore con CFA, può essere opportuno tenere conto, in funzione del guadagno desiderato, della diminuzione della frequenza di taglio superiore causata dalla resistenza R_b del buffer.

Esempio 9.1

Si prenda in considerazione l'op-amp a reazione di corrente CLC449 previsto per varie realizzazioni, come filtri attivi in alta frequenza, apparecchiature elettromedicali, driver ad alta velocità nei convertitori analogico-digitali e come buffer nei convertitori digitale-analogici.
Le specifiche forniscono, fra gli altri dati, uno slew rate di 2500 V/ μs e una frequenza di taglio di 1,1 GHz per un guadagno A_{CL}=2.
L'alimentazione può essere duale a ±5 V o singola a 10 V. Per realizzare un amplificatore non invertente – figura 9.4 – con banda passante

estesa fino a 1,1 GHz e guadagno pari a 2, il costruttore indica per R_f il valore:

$$R_f = 250\ \Omega$$

Nell'ipotesi di fare ricorso alla configurazione non invertente dell'op-amp per la quale è:

$$A_{CL} = 1 + (R_f/R_i)$$

e quindi anche:

$$R_i = R_f/(A_{CL} - 1) \qquad [9.15]$$

Per $A_{CL} = 2$ è necessario attribuire alla resistenza R_i il valore:

$$R_i = 250 / (2 - 1) = 250\ \Omega$$

La resistenza R, come già si è visto per i VFA, fissa il valore della resistenza di ingresso dello stadio.
La figura 9.4 riporta l'amplificatore con op-amp appena dimensionato, mentre la figura 9.5 fornisce il diagramma della risposta in frequenza ottenuto in simulazione: a – 3 dB, indicati dalla posizione reciproca dei marker orizzontali *c* e *d*, il marker verticale *a* indica una frequenza di taglio di 1,1 GHz.
La distanza fra il livello 0 dB (livello del segnale di ingresso) e il tratto orizzontale della curva è l'ordinata $Yd = 5{,}933$ dB a cui corrisponde un guadagno $A_{CL} = 1{,}98$.
La figura 9.6 riporta quindi l'andamento delle forme d'onda in ingresso e in uscita: è $V_i = 1$ V da picco a picco e $V_o = 2$ V (1,993 V) da picco a picco. Più in generale, per guadagni inferiori a 5 le note tecniche forniscono, per il calcolo del valore da attribuire alla R_f, l'espressione:

$$R_f = 340 - 45\ A_{CL}$$

Per valori del guadagno maggiori di 5 il minimo valore raccomandato per R_f è di 100 Ω.
La figura 9.7 riporta la risposta, ricavata in simulazione al computer, di un amplificatore realizzato con l'op-amp a reazione di corrente CLC404 in configurazione non invertente per tre diversi valori del guadagno ottenuti attribuendo alla R_f tre distinti valori desunti, appunto in funzione di A_{CL}, dall'espressione (fornita dal costruttore):

$$R_f = 680 - 30\ A_{CL}$$

e ricavando quindi il valore della resistenza R_i tramite la [9.15]. Si noti come questo diagramma sia del tutto simile al diagramma teorico riportato nella figura 9.3 *b*.
Il CLC404 presenta uno slewrate di 2600 V/ µs e una banda passante ai grandi segnali ($V_o = 5$ V da picco a picco) di 165 MHz.
All'aumentare del guadagno si nota, in effetti, una contrazione della banda passante che si estende fino a 234 MHz per $A_{CL} = 6$ dB, fino a 181 MHz per $A_{CL} = 15{,}5$ dB e fino a 162 MHz per $A_{CL} = 21$ dB, ma questa contrazione, pur potendo apparire notevole nel passaggio da $A_{CL} = 6$ dB a $A_{CL} = 21$ dB, rimane comunque contenuta in termini applicativi del CFA (si noti, fra l'altro, che fra le due curve estreme ci sono 15 dB di differenza).
Le stesse considerazioni valgono quindi per la configurazione invertente dei CFA per i quali, al fine di evitare l'insorgere di oscillazioni, il costrut-

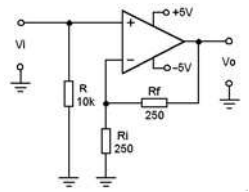


Figura 9.4 *Stadio amplificatore con CFA in configurazione non invertente con guadagno pari a 2.*

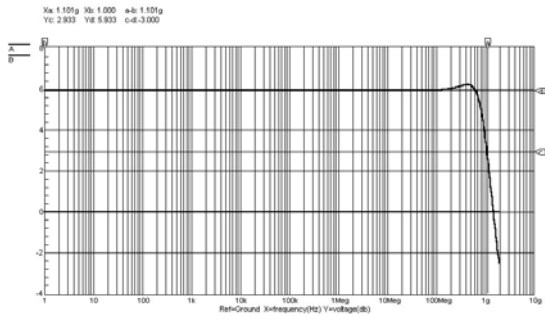


Figura 9.5 *Risposta in frequenza dell'amplificatore di cui alla figura 9.4. Per un guadagno pari a 2 la frequenza di taglio è di 1,1 GHz in assenza di carico e di 938 MHz con un carico da 100 Ω.*

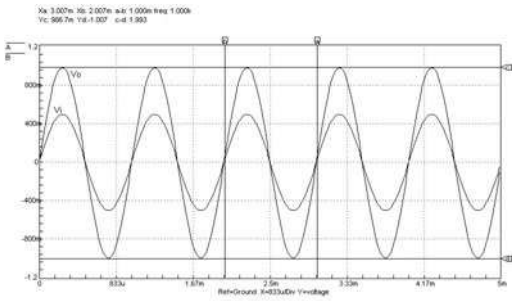


Figura 9.6 *Andamento delle forme d'onda in ingresso e in uscita nello stadio amplificatore di cui alla figura 9.4.*

tore suggerisce di inserire una resistenza fra il pin non invertente e massa di valore in genere non superiore a 100 Ω.

9.4 L'INSEGUITORE DI TENSIONE CON CFA

Un'altra differenza fra gli op-amp a reazione di tensione e gli op-amp a reazione di corrente si coglie nella realizzazione del voltage follower.
La figura 9.8 *a* riporta il voltage follower realizzato con il CFA, mentre la figura 9.8 *b* riporta lo stesso circuito realizzato con il VFA.
L'unica differenza risiede, come è evidente, nella resistenza *R* posta sull'anello di reazione del voltage follower realizzato con l'amplificatore a reazione di corrente. Questa resistenza, il cui valore è normalmente fornito dalle Case, è necessaria al fine di evitare l'innesco di oscillazioni.

Esempio 9.2

Qui si dimensiona uno stadio amplificatore invertente in alternata – figura 9.9 – con guadagno A_{CLi} = 10 per la banda passante 1 MHz ÷ 100 MHz a –3 dB, utilizzando l'amplificatore operazionale a reazione di corrente AD8002. Per questo, il Costruttore, Analog Devices, fornisce la banda passante di 600 MHz a –3 dB con guadagno unitario e uno slewrate di oltre 1000 V/ µs.
Nei fogli tecnici relativi all'op-amp si legge che, per un guadagno 10, è necessario porre R_f = 499 Ω.
In simulazione si è posto R_f = 500 Ω. Per un guadagno 10, essendo $A_{CLi} = R_f/R_i = 10$, si ha per R_i:

$$R_i = 500 / 10 = 50\ \Omega$$

Non rimane che da dimensionare le capacità dei tre condensatori. Per il condensatore di ingresso vale l'espressione già nota:

$$C_I \geq 0{,}159 / (f_L \cdot R_i)$$

Per f_L = 1 MHz e R_i = 50 Ω, si ricava:

$$C_I \geq 0{,}159/(10^6 \times 50) = 3{,}18\ \text{nF} \Rightarrow C_I = 3{,}3\ \text{nF}$$

Le so tutte!!!

Risposta al quesito codice LST24509

Il circuito è un filtro passa alto passivo con frequenza di taglio pari a 1/(2πRC) ≅ 100KHz il che significa che tutti i segnali di frequenza superiore ad 1MHHz (una decade sopra la frequenza di taglio) passano inalterati. I segnali aventi frequenza inferiore a 10KHz (una decade prima della frequenza di taglio) verranno derivati. Detto questo risulta piuttosto evidente che applicando un segnale triangolare di frequenza 1KHz (T=1ms) questo viene derivato e l'uscita sarà un'onda quadra di frequenza 1KHz per cui la prima affermazione è vera e , di conseguenza la quarta affermazione è falsa. Essendo un circuito passivo il segnale verrà sicuramente attenuato per cui se il segnale di ingresso ha una ampiezza di 2Vpp, il segnale in uscita avrà sicuramente una ampiezza minore (la seconda affermazione è dunque falsa).
Il circuito non contiene elementi non lineari quindi l'uscita è ancora un segnale periodico e la terza affermazione è vera. La quinta affermazione è falsa in quanto il condensatore in serie all'ingresso blocca solo l'eventuale componente continua del segnale.

Il vincitore di SETTEMBRE 2005 (Vincitore LST24408)
I nostri complimenti a Vanni Roldi di Alessandria
che vince un abbonamento a Fare Elettronica!

Per la capacità del condensatore C_2 che limita superiormente la banda passante e va quindi calcolata in funzione della f_H = 100 MHz, si ha l'espressione altrettanto nota:

$$C_2 \leq 0{,}159/(f_H \cdot R_f)$$

si ha pertanto:

$$C_2 \leq 0{,}159/(100 \times 10^6 \times 500) = 3{,}16 \text{ pF} \Rightarrow C_2 = 2{,}8 \text{ pF}$$

Per C_o, indicando sempre con f_L la frequenza di taglio inferiore, dev'essere:

$$C_o \geq 0{,}159/(f_L \cdot R_L)$$

si ha quindi:

$$C_o \geq 0{,}159/(10^6 \times 5000) = 31{,}8 \text{ pF} \Rightarrow C_3 = 82 \text{ pF}$$

La figura 9.10 riporta le forme d'onda in ingresso e in uscita. L'ampiezza da picco a picco della V_o, indicata dalla distanza (*c* – *d*), è pari a 3,967 V. Poichél'amplificatore è stato pilotato con un segnale di 400 mV da picco a picco, il guadagno è: V_o/V_i = 3967/400 = 9,91. La distanza fra i marker verticali *a* e *b*, ossia (*a* – *b*), indica la frequenza del segnale pilota di 10 MHz. Data la configurazione, l'uscita, rispetto all'ingresso, è sfasata di 180°nel campo di frequenze in cui la curva di risposta in frequenza – figura 9.11 – è piatta, ossia laddove il guadagno è massimo e pari a 20 dB. A –3 dB si ha una frequenza di taglio inferiore f_L = 1,013 MHz indicata dalla posizione del marker verticale *b*, e una frequenza di taglio superiore f_H = 100 MHz indicata dalla posizione del marker verticale *a*.
L'ordinata *Yd* = 19,92 dB indica il guadagno dello stadio che è di poco inferiore alla richiesta (20 dB).
L'esempio appena riportato, così come il precedente, dimostra che il dimensionamento di uno stadio amplificatore con CFA si esegue con gli stessi criteri adottati per i VFA prestando solo maggiore attenzione ai suggerimenti delle Case che, fra l'altro, forniscono sempre i valori ottimali da attribuire alle diverse resistenze dell'anello di reazione in funzione del guadagno desiderato.
È comunque evidente, con riferimento ai CFA, il grande vantaggio di disporre di dispositivi di amplificazione ad elevato slewrate e nei quali l'ampiezza della banda passante, entro ampi margini, è svincolata dal guadagno.

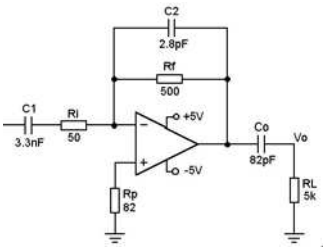


Figura 9.9 *Amplificatore in configurazione invertente a guadagno 10 e banda passante (1 ÷ 100) MHz*

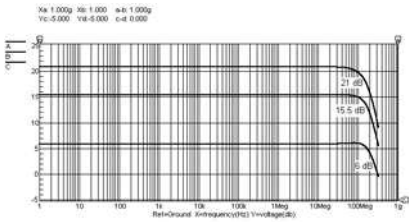


Figura 9.7 *Curve di risposta in frequenza del CFA CLC404 per diversi valori del guadagno*

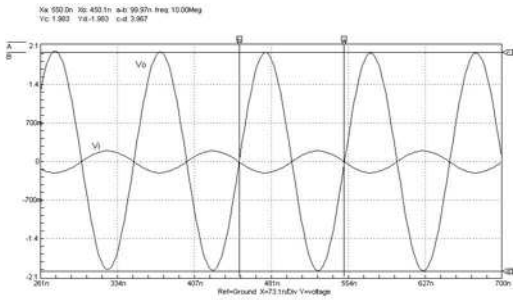


Figura 9.10 *Forme d'onda in ingresso e in uscita nello stadio amplificatore di cui alla figura 10.9. La configurazione invertente sfasa di 180° l'uscita rispetto all'ingresso nell'intorno del centrobanda*

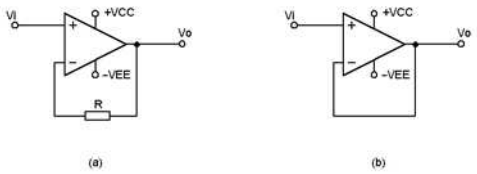


Figura 9.8 *Nell'inseguitore ditensione con CFA, (a), è necessario disporre lungo il canale di reazione una resistenza R, mentre nei VFA, (b) la connessione uscita-ingresso invertente è diretta.*

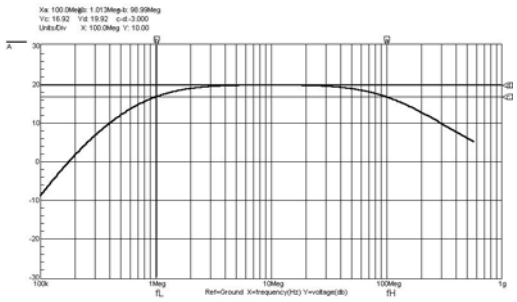


Figura 9.11 *Risposta in frequenza dell'amplificatore di cui alla figura 9.9. La banda passante coincide quasi perfettamente con i presupposti del progetto*

PS1503SB

Alimentatore 0-15Vdc / 0-3A

Uscita stabilizzata singola 0 -15Vdc con corrente massima di 3A. Limitazione di corrente da 0 a 3A impostabile con continuità.
Due display LCD con retroilluminazione indicano la tensione e la corrente erogata dall'alimentatore. Contenitore in acciaio, pannello frontale in plastica. Colore: bianco/grigio; peso: 3,5 Kg.

PS1503SB 62,00

PS3010

Alimentatore 0-30Vdc/0-10A

Alimentatore stabilizzato con uscita singola di 0 - 30Vdc e corrente massima di 10A. Limitazione di corrente da 0 a 10A impostabile con continuità. Due display indicano la tensione e la corrente erogata dall'alimentatore. Contenitore in acciaio, pannello frontale in plastica. Colore: bianco/grigio; peso: 12 Kg.

PS3010 216,00

PS3020

Alimentatore 0-30Vdc/0-20A

Alimentatore stabilizzato con uscita singola di 0-30Vdc e corrente massima di 20A. Limitazione di corrente da 0 a 20A impostabile con continuità. Due display indicano la tensione e la corrente erogata dall'alimentatore.
Contenitore in acciaio, pannello frontale in plastica. Colore: bianco/grigio; peso: 17 Kg.

PS3020 330,00

PS230210

Alimentatore con uscita duale

Alimentatore stabilizzato con uscita duale di 0-30Vdc per ramo con corrente massima di 10A. Ulteriore uscita stabilizzata a 5Vdc. Quattro display LCD indicano contemporaneamente la tensione e la corrente erogata da ciascuna sezione; possibilità di collegare in parallelo o in serie le due sezioni. Contenitore in acciaio, pannello frontale in plastica. Colore: bianco/grigio; peso: 20 Kg.

PS230210 616,00

con tecnologia SWITCHING

LA TECNOLOGIA SWITCHING CONSENTE DI OTTENERE UNA NOTEVOLE RIDUZIONE DEL PESO ED UN ELEVATISSIMO RENDIMENTO ENERGETICO DELL'APPARECCHIATURA.

ALIMENTATORI DA LABORATORIO

Alimentatore stabilizzato con uscita duale di 0-30Vdc per ramo con corrente massima di 3A. Ulteriore uscita stabilizzata a 5Vdc con corrente massima di 3A. Quattro display LCD indicano contemporaneamente la tensione e la corrente erogata da ciascuna sezione; limitazione di corrente 0÷3A impostabile indipendentemente per ciascuna uscita. Possibilità di collegare in parallelo o in serie le due sezioni. Peso: 11,6 Kg.

PS23023 252,00

PS23023

Alimentatore 2x0-30V/0-3A 1x5V/3A

Alimentatore stabilizzato con uscita singola di 0-30Vdc e corrente massima di 3A. Limitazione di corrente da 0 a 3A impostabile con continuità.
Due display LCD indicano la tensione e la corrente erogata dall'alimentatore. Contenitore in acciaio, pannello frontale in plastica. Colore: bianco/grigio. Peso: 4,9 Kg.

PS3003 125,00

PS3003

Alimentatore 0-30Vdc/0-3A

Alimentatore stabilizzato con uscita singola di 0-50Vdc e corrente massima di 5A. Limitazione di corrente da 0 a 5A impostabile con continuità. Due display indicano la tensione e la corrente erogata dall'alimentatore.
Contenitore in acciaio, pannello frontale in plastica. Colore: bianco/grigio. Peso: 9,5 Kg.

PS5005 225,00

PS5005

Alimentatore 0-50Vdc/0-5A

Alimentatore da banco stabilizzato con tensione di uscita selezionabile a 3 - 4.5 - 6 - 7.5 - 9 - 12Vdc e selettore on/off.
Bassissimo livello di ripple con LED di indicazione stato. Protezione contro corto circuiti e sovraccarichi. Peso: 1,35 Kg.

PS2122LE 18,00

PS2122LE

Alimentatore da banco 1,5A

Alimentatore stabilizzato da laboratorio in tecnologia switching con indicazione delle funzioni mediante display multilinea. Tensione di uscita regolabile tra 0 e 20Vdc con corrente di uscita massima di 10A. Soglia di corrente regolabile tra 0 e 10A. Il grande display multifunzione consente di tenere sotto controllo contemporaneamente tutti i parametri operativi.
Caratteristiche: *Tensione di uscita: 0-20Vdc; limitazione di corrente:0-10A; ripple con carico nominale: inferiore a 15mV (rms); display: LCD multilinea con retroilluminazione; dimensioni: 275 x 135 x 300 mm; peso: 3 Kg.*

PSS2010 265,00

PSS2010

Alimentatore Switching 0-20Vdc/0-10A

PSS4005

Alimentatore Switching 0-40Vdc/0-5A

Alimentatore stabilizzato da laboratorio in tecnologia switching con indicazione delle funzioni mediante display multilinea. Tensione di uscita regolabile tra 0 e 40Vdc con corrente di uscita massima di 5A. Soglia di corrente regolabile tra 0 e 5A.
Caratteristiche: *tensione di uscita: 0-40Vdc; limitazione di corrente: 0-5A; ripple con carico nominale: inferiore a 15 mV (rms); display: LCD multilinea con retroilluminazione; dimensioni: 275 x 135 x 300 mm; peso: 3 Kg.*

PSS4005 265,00

ALIMENTATORI A TENSIONE FISSA

PS1303

Alimentatore 13,8Vdc/3A

Alimentatore stabilizzato con uscita singola di 13,8 Vdc in grado di erogare una corrente massima di 3A (5A di picco). Il circuito di alimentazione a 220 Vac è protetto tramite fusibile mentre l'uscita dispone di protezione da cortocircuiti. Contenitore in acciaio. Colore: bianco/grigio; peso: 1,7 Kg.

PS1303 26,00

PS1310

Alimentatore 13,8Vdc/10A

Alimentatore stabilizzato con uscita singola di 13,8 Vdc in grado di erogare una corrente massima di 10A (12A di picco). Il circuito di alimentazione a 220 Vac è protetto tramite fusibile mentre l'uscita dispone di protezione da cortocircuiti. Contenitore in acciaio. Colore: bianco/grigio; peso: 4 Kg.

PS1310 43,00

PS1320

Alimentatore 13,8Vdc/20A

Alimentatore stabilizzato con uscita singola di 13,8 Vdc in grado di erogare una corrente massima di 20A (22A di picco). Il circuito di alimentazione a 220 Vac è protetto tramite fusibile mentre l'uscita dispone di protezione da cortocircuiti. Contenitore in acciaio. Colore: bianco/grigio; peso: 6,7 Kg.

PS1320 95,00

PS1330

Alimentatore 13,8Vdc/30A

Alimentatore stabilizzato con uscita singola di 13,8 Vdc in grado di erogare una corrente massima di 30A (32A di picco). Il circuito di alimentazione a 220 Vac è protetto tramite fusibile mentre l'uscita dispone di protezione da cortocircuiti. Contenitore in acciaio. Colore: bianco/grigio; peso: 9,3 Kg.

PS1330 140,00

Tutti i prezzi si intendono IVA inclusa.

FUTURA ELETTRONICA

Via Adige, 11 - 21013 Gallarate (VA)
Tel. 0331/799775 - *www.futuranet.it*

Maggiori informazioni su questi prodotti e su tutte le altre apparecchiature distribuite sono disponibili sul sito *www.futuranet.it* tramite il quale è anche possibile effettuare acquisti on-line.

Seconda parte
n° 245 - Novembre 2005
Utilizzare le CPLD XC9500 e l'ambiente di sviluppo Xilinx

Terza parte
n° 246 - Dicembre 2005
Sistema hardware di sviluppo

Quarta parte
n° 247 - Gennaio 2006
Interfaccia per la programmazione in-system

CPLD by Example

Nella puntata precedente abbiamo trattato la struttura dei dispositivi XC9500 e abbiamo visto come installare e utilizzare l'ambiente di progetto Xilinx ISE WebPack. In questa puntata presentiamo il circuito general-purpose che servirà da base per sviluppare gli esempi e i progetti che verranno trattati in seguito. Contiene una CPLD Xilinx montata su zoccolo e componenti del tipo a inserzione per facilitare il montaggio anche ai meno esperti. Vedremo poi alcuni esempi applicativi e una introduzione sugli operatori Verilog. Come già fatto in precedenza, conserviamo la suddivisione in moduli, così da raggruppare e focalizzare meglio gli argomenti trattati.

MODULO 4
CIRCUITO BASE

Lo schema elettrico del circuito base è riportato in figura 1 e il master del relativo circuito stampato è descritto in figura 2. Come possiamo notare dallo schema, si è cercato di inserire solo i componenti strettamente necessari per il funzionamento della CPLD (IC1) e di mettere a disposizione la maggior parte dei pin di I/O di questa ai due connettori a striscia J3 e J4, insieme alle alimentazioni e alla massa. Questa scelta è stata fatta in previsione di poter collegare alla piastrina base alcuni circuiti di tipo piggy-back, da inserire ai connettori suddetti, specializzati a svolgere funzioni particolari. Ad esempio, in una prossima puntata presenteremo una piastrina con display a quattro cifre. Come possiamo immaginare, le possibilità di espansione sono limitate solo dalla fantasia; a titolo di esempio, possiamo pensare di interfacciare motorini passo-passo, display LCD, interfacce seriali, memorie, eccetera.

Nello schema è presente un oscillatore (IC2) da 1,843200 Mhz. È consigliabile montarlo su zoccolo, in modo da poterlo sostituire agevolmente nel caso si voglia utilizzare una diversa sorgente di frequenza. Nelle altre prove abbiamo utilizzato oscillatori da 4,096 Mhz fino a 100 Mhz. I dispositivi di questo tipo sono realizzati in case metallico e hanno una piedinatura standard adatta a zoccoli da 14 pin dual-in-line. L'alimentazione è a 5V e il livello del segnale di uscita è TTL/CMOS compatibile.

All'accensione, il reset alla CPLD viene fornito dal gruppo R-C costituito da R4 e C2.

Per la programmazione *in-system*, si deve collegare la relativa interfaccia JTAG al connettore J2. Nella prossima puntata presenteremo lo schema elettrico dell'interfaccia.

L'alimentazione richiesta dalla CPLD è a +3.3V, ed è ottenuta con il regolatore serie IC4.

Per alimentare l'oscillatore è stato aggiunto anche il regolatore IC3, che fornisce la tensione a +5V. Le due tensioni regolate, insieme a quella proveniente dall'alimentatore esterno (+VDC), sono riportate al connettore J4, per gli eventuali utilizzatori esterni.

Infine, non potevano mancare alcuni led, da DL1 a DL3, che possono essere utili per monitorare il "battito cardiaco" del nostro circuito.

Sistema hardware di sviluppo

di Agostino Rolando
a.rolando@farelettronica.com

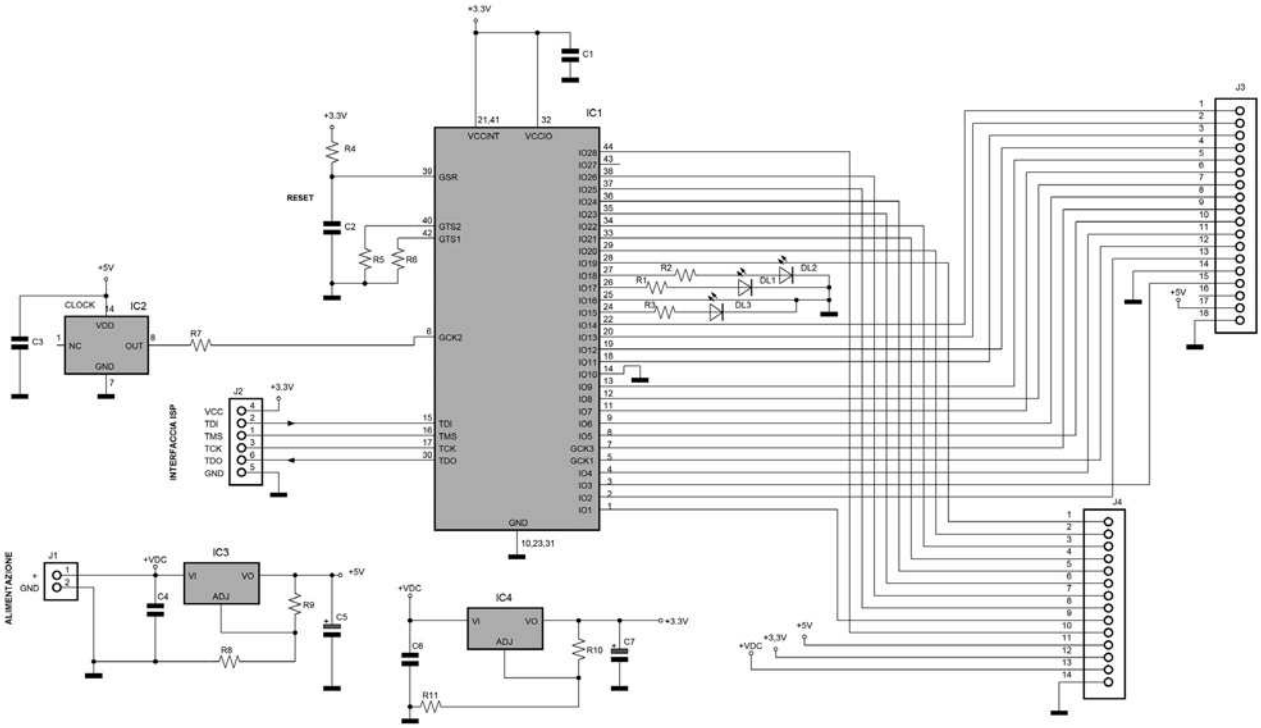


Figura 1 Schema elettrico

MODULO 5

Passiamo ora ad esaminare alcuni esempi applicativi in codice Verilog.

ESEMPIO 1: CATTURARE UN IMPULSO

Spesso si presenta la necessità di dover rilevare un segnale impulsivo di durata anche molto breve ed eventualmente affetto da spikes, come ad esempio è il caso di un segnale generato da un pulsante. Nell'esempio che segue supponiamo di aver collegato un pulsante ad un ingresso della CPLD, con chiusura verso massa e dotato di una resistenza di pull-up verso l'alimentazione a +3.3V.

In questi casi è conveniente campionare il segnale con un clock sufficientemente veloce, in modo da poter rilevare la sua transizione alto-basso; proprio quello che fa il codice che segue. La funzione svolta è l'equivalente di un circuito monostabile non retriggerabile: in sostanza, viene rilevata la chiusura del pulsante e, di conseguenza, si genera un impulso squadrato di durata prefissata, pari a 256 colpi di clock. Per rendere visibile l'evento inviamo l'uscita ad uno dei led disponibili.

```
module monostab (CLK,RES,INTRIG,LED);
  // I/O
  input CLK;
  input RES;
  input INTRIG; // trigger di input,
  attivo low
  output LED;
  // Segnali interni
  reg [7:0] conta;
  reg contenab;
  wire tcont;
  // Implementazione
  assign tcont = (conta==8'b1);
```

```
  // Circuito monostabile
  always @(posedge CLK or negedge RES)
  begin
    if (!RES)
      contenab <= 1'b0;
    else
      if (!INTRIG)
        contenab <= 1'b1;
      else
       if (tcont)
         contenab <= 1'b0;
  end
  // Contatore
  always @ (posedge CLK or negedge RES)
  begin
    if (!RES)
      conta <= 8'b0;
    else
      if (!contenab)
        conta <= 8'b0;
      else
        conta <= conta + 1;
  end
  assign LED = contenab;
endmodule
```

Nel codice, *INTRIG* è il segnale di ingresso, contenab è una variabile che costituisce l'abilitazione al conteggio per il contatore *conta*, mentre *tcont* rappresenta il "terminal count", segnale che viene posto a "1" quando il contatore raggiunge il valore massimo (255). *LED* è il segnale di uscita.

ESEMPIO 2: MULTIPLEXER

I multiplexer possono essere modellati in molti modi differenti; ne vediamo alcuni tra i più utilizzati. Il primo consiste nell'impiegare una cascata di istruzioni di tipo *if..else*, il secondo fa uso dell'istruzione *case*.
Come prima applicazione, analizziamo un multiplexer da 4 ingressi e un'uscita (figura 6). Al modulo vengono passati i segnali *inA÷inD* più il selettore a due bit *sel*. L'uscita è indicata con Y:

```
module mux_4_1 (sel, inA, inB, inC, inD, Y);
  // I/O
  input [1:0] sel;
  input inA, inB, inC, inD;
  output Y;
    reg Y;
  // Implementazione
  // .. con l'istruzione if
  always @ (sel or inA or inB or inC or inD)
  begin
    if (sel == 2'b00)
      Y=inA;
    else
      if (sel == 2'b01)
        Y=inB;
      else
        if (sel == 2'b10)
          Y=inC;
        else
          if (sel == 2'b11)
            Y=inD;
  end

// ..con l'istruzione case
  always @(sel or inA or inB or inC or inD)
  begin
    case(sel)
      2'b00: Y=inA;
      2'b01: Y=inB;
      2'b10: Y=inC;
      2'b11: Y=inD;
    default: Y=inA;
    endcase
  end
endmodule
```

Il codice, se viene compilato ad esempio per una CPLD 9536XL, impegna una singola macrocella.
In questa applicazione vengono utilizzati sette pin, denominati inA, inB, inC, inD, sel<0>, sel<1> e Y.
Il progetto è puramente combinatorio e si traduce in quattro termini di prodotto; infatti, se diamo un'occhiata al file di report del processo *Fitter*, possiamo evidenziare che le equazioni logiche con cui viene implementato sono le seguenti:

```
Y = inD & sel<1> & sel<0>
  # sel<1> & !(sel<0>) & inC
  # !(sel<1>) & sel<0> & inB
  # !(sel<1>) & !(sel<0>) & inA;
```

La schematica equivalente è quella di figura 7.

Ritorna a Udine
la fiera dell'elettronica & del radioamatore

Con oltre 150 espositori e tanta informatica ai prezzi più bassi d' Italia

MOSTRA MERCATO

10-11
DICEMBRE 2005

ORARIO AL PUBBLICO:
09.00 - 18.00

prevendita biglietti dalle ore 08.30

Informazioni: 337 676719
E-mail: info@compendiofiere.it
Intero € 6,00 ridotto € 4,00
Padiglioni 5 e 6 - Ingresso Ovest
Parcheggio gratuito

Udine Fiere
Quartiere Fieristico Udinese - Torreano di Martignacco (UD) - www.udinefiere.it

- COMPUTER
- ELETTRONICA
- RADIANTISMO
- EDITORIA
- TELEFONIA
- TV-SATELLITARE
- HOBBISTICA

Organizzazione :
COMPUTER & RADIO
Compendio Fiere S.r.l.
Vi aspetta a Cerea (VR)
il 14 e 15 gennaio 2006

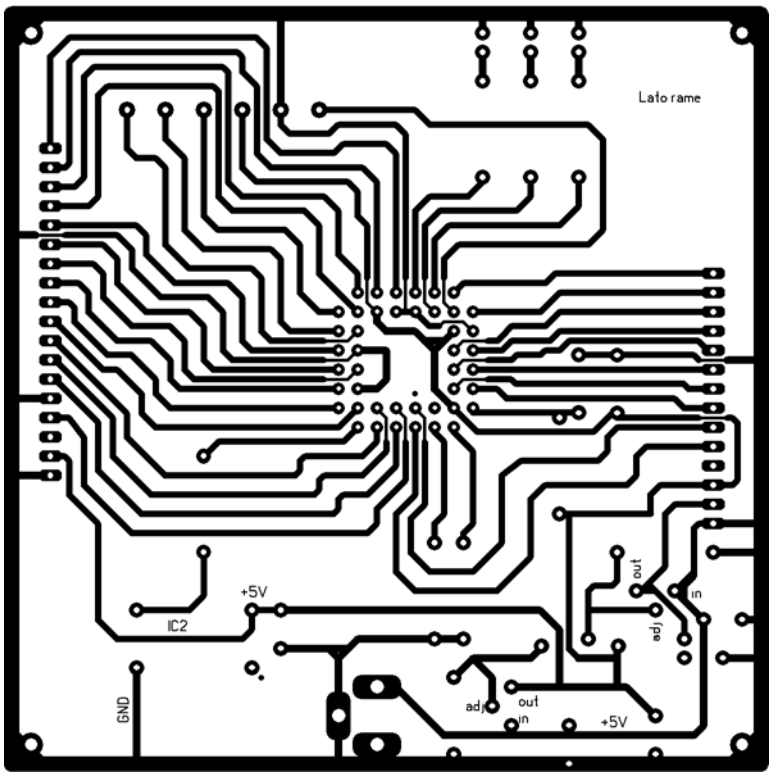


Figura 2 *Circuito stampato in scala 1:1 (lato rame)*

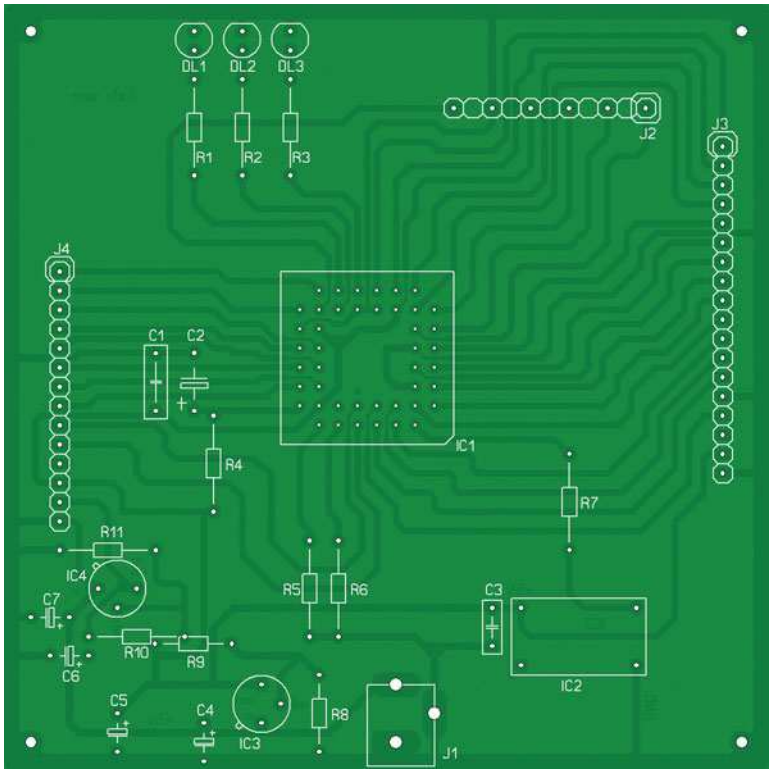


Figura 2a *Piano di montaggio*

Elenco componenti	
Sigla	**Valore**
R1÷R3, R9	330 Ω 1/4 W
R4÷R6	4,7 KΩ 1/4 W
R7	100 Ω 1/4 W
R8	1 KΩ 1/4 W
R10	820 Ω 1/4 W
R11	1,2 KΩ 1/4 W
C1, C3, C4, C6	0,1 µF 50 V ceramico
C2	4,7 µF 50 V elettrolitico
C5, C7	33 µF 50 V elettrolitico
DL1÷DL3	Diodo LED 5 mm
IC1	CPLD XC9572XL-PC44C, package PLCC-44
IC2	Oscillatore 1,843200 MHz
IC3, IC4	LM317H, contenitore TO39 (con dissipatore)
J1	Connettore da cs (tipo plug)
J2	Connettore 6 pin M (passo 5,08)
J3	Connettore 18 pin, M (passo 2,54)
J4	Connettore 14 pin, M (passo 2,54)

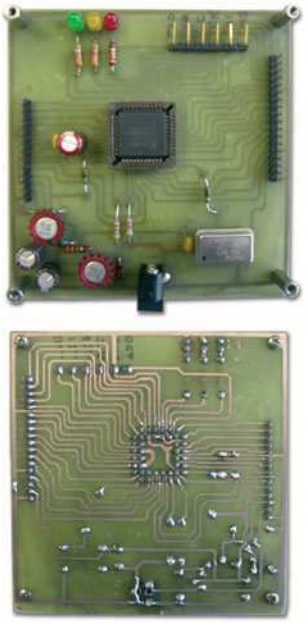


Figura 3 *Prototipo lato componenti e lato saldature*

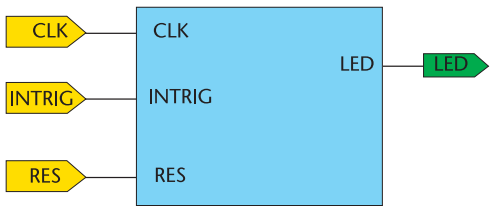


Figura 4 Circuito monostabile

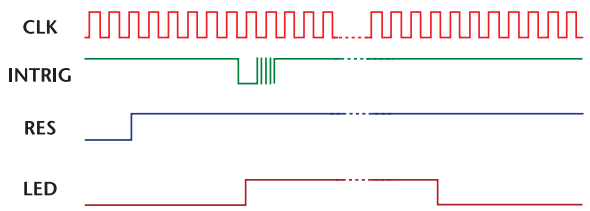


Figura 5 Temporizzazioni per l'esempio N.1

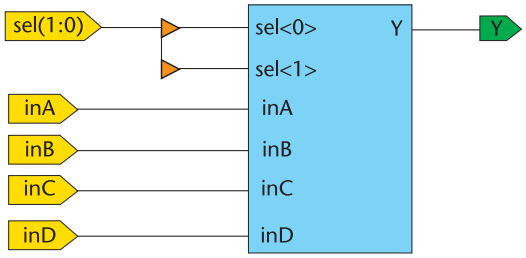


Figura 6 Schema a blocchi del multiplexer 4-1

MULTIPLEXER A 8 INGRESSI

Vediamo ora un altro esempio, simile al predecente ma leggermente più complesso in quanto sono previsti otto ingressi e tre selettori. In questo caso, applichiamo nuovamente il costrutto case e otteniamo il codice seguente:

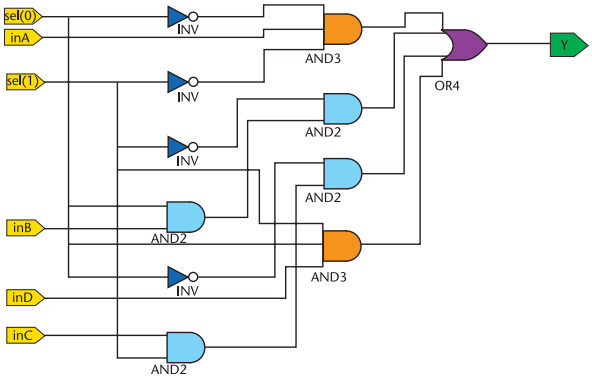


Figura 7 Implementazione a livello di gate del multiplexer 4-1

```
module  mux_8_1(sel,inA0,inA1,inA2,inA3,inA4,
  inA5,inA6,inA7,Y);
  // I/O
  input [2:0] sel;
  input  inA0, inA1, inA2, inA3, inA4,
inA5, inA6, inA7;
  output  Y;
  reg  Y;
  // Implementazione
  always@(sel or inA0 or inA1 or inA2 or
inA3 or inA4 or inA5 or inA6 or inA7)
   begin
     case(sel)
       3'b000 : Y=inA0;
       3'b001 : Y=inA1;
       3'b010 : Y=inA2;
       3'b011 : Y=inA3;
       3'b100 : Y=inA4;
       3'b101 : Y=inA5;
       3'b110 : Y=inA6;
       3'b111 : Y=inA7;
     default : Y=inA0;
     endcase
   end
endmodule
```

Se guardiamo il sommario delle risorse utilizzate (riportato nel riquadro 1), possiamo notare che, questa volta, il multiplexer utilizza ancora una sola macrocella, ma 8 termini di prodotto. L'implementazione è la seguente:

```
Y = inA7 & sel<2> & sel<1> & sel<0>
   # sel<2> & !(sel<1>) & inA5 & sel<0>
   # !(sel<2>) & inA3 & sel<1> & sel<0>
   # !(sel<2>) & inA1 & !(sel<1>) & sel<0>
   # !(sel<2>) & inA0 & !(sel<1>) & !(sel<0>)
   # sel<2> & sel<1> & !(sel<0>) & inA6
   # sel<2> & !(sel<1>) & !(sel<0>) & inA4
   # !(sel<2>) & inA2 & sel<1> & !(sel<0>);
```

La schematica equivalente al codice risulta piuttosto intricata. Ci limitiamo a riportare lo schema a blocchi a livello *top* (figura 8).

ESEMPIO 3: ENCODER

Nella puntata precedente abbiamo già visto un circuito applicativo di questo genere. Per quanto riguarda il suo funzionamento, possiamo notare che, dato un certo insieme di segnali di ingresso in codifica one-hot (in cui un solo ingresso alla volta assume il valore "1" ad un dato istante), l'encoder genera un set di uscite in codifica binaria, impiegando quindi un

[riquadro 1]

```
****  Resource Summary  ****
Macrocells    Product       Terms        Registers      Pins    Function Block
Used          Used          Used         Used           Used
1/72  (1%)    8/360  (2%)   0/72  (0%)   12/34  (35%)   11/216  (5%)
```

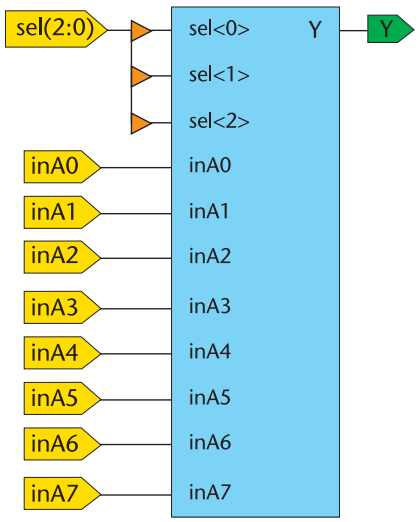


Figura 8 *Schema a blocchi del multiplexer 8-1*

Inputs								Outputs		
A7	A6	A5	A4	A3	A2	A1	A0	Y2	Y1	Y0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

Tabella 1 *Tavola di verità per l'encoder 8:3*

numero di pin inferiore rispetto agli ingressi. Come esempio, riportiamo la tavola di verità di un encoder da 8 ingressi e 3 uscite (tabella 1).

Il codice che descrive questa tabella può essere messo in atto o con l'istruzione ***if*** o con un ***case*** o con un loop ***for***.
Anche in questa situazione il *case* (meglio *casex*) è più conciso e leggibile rispetto agli altri costrutti e ciò è tanto più vero quanto più numerosi sono gli ingressi del modulo.
In questa situazione, i possibili valori di ingresso sono $2^8 = 256$. Per abbracciare tutti i casi utilizziamo la notazione "X", che indica sinteticamente le situazioni "*don't care*".

Codifica

La codifica dell'encoder è la seguente:

```
module encoder_8_3(A,Y);
  // I/O
  input [7:0] A;
  output [2:0] Y;
  reg [2:0] Y;
  // Implementazione
  //  A. Con l'istruzione if
  always @(A)
  begin
    if      (A == 8'b00000001) Y=0;
    else if (A == 8'b00000010) Y=1;
    else if (A == 8'b00000100) Y=2;
    else if (A == 8'b00001000) Y=3;
    else if (A == 8'b00010000) Y=4;
    else if (A == 8'b00100000) Y=5;
    else if (A == 8'b01000000) Y=6;
    else if (A == 8'b10000000) Y=7;
    else                       Y=3'bX;
  end
  //  B. Con l'istruzione casex
  always @(A)
  begin
    casex(A)
      8'b00000001: Y=0;
      8'b00000010: Y=1;
      8'b00000100: Y=2;
      8'b00001000: Y=3;
      8'b00010000: Y=4;
      8'b00100000: Y=5;
      8'b01000000: Y=6;
      8'b10000000: Y=7;
      default: Y=3'bX;
    endcase
  end
  //  C. Con il loop for
  reg [7:0] test;
  integer N;
```

```
  always @(A)
  begin
    test = 8'b00000001;
    Y = 3'bX;
    for (N=0; N < 8; N = N+1)
      begin
        if (A == test)
          Y = N;
          test = test << 1;
     end
   end
endmodule
```

ESEMPIO 4: DECODER

Il decoder effettua la funzione inversa dell'encoder, cioè da un set di ingressi in codifica binaria produce un set di uscite di cui una sola alla volta è attiva. La tavola di verità per un decoder da 3 ingressi e 8 uscite è come da tabella 2. Notiamo che le relazioni sono esattamente reciproche, rispetto alla tabella 1.

Anche in questo caso possiamo modellare la funzione con i tre costrutti *if*, *case* o *for*.

Inputs			Outputs							
A0	A1	A2	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

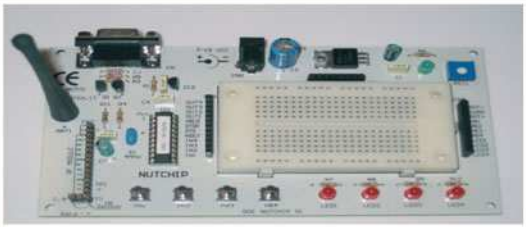
Tabella 2 *Tavola di verità per l'encoder da 3 a 8*

```
module decoder_3_8(A,Y);
  // I/O
  input [2:0] A;
  output [7:0] Y;
```


Con Artek hai l'elettronica a portata di un click.
Esplora il nostro sito, ogni mese scoprirai le novità dell'Elettronica, il mondo dei Microcontrollori, nuovi sensori e strumenti per progetti di Robotica.
Inoltre strumenti di misura digitali professionali interfacciati al pc per il laboratorio

Artek ti offre un modulo per programmare i PIC Microchip con funzioni di debug e test a soli 62 Euro IVA compresa.
Visita il nostro sito per sapere di più sul C-Project C-170.

Costruisci un Robot con il BASIC Stamp
il microcontrollore più famoso e diffuso fra gli appassionati di Robotica per la sua semplicità d'uso e la vasta gamma di accessori



BoeNut 01
stazione di lavoro completa per **Nutchip**

Strumenti digitali di misura su porta USB
- due canali
- 12 bit di risoluzione
- ingresso fino a 100 MHZ

Puoi avere questa mini-telecamera **a colori,** completa di **ricevitore** e **microfono** ad un prezzo che non ha eguali !

La nostra vetrina è on-line all'indirizzo www.artek.it : puoi controllare le caratteristiche, i prezzi e ordinare da subito ciò che ti occorre. Puoi contattarci con una e-mail a diramm@artek.it inviando un fax allo 0542 688405 oppure chiamando i nostri uffici allo 0542 643192
dalle 9 alle 13:30 e dalle 14:30 alle 18 dal Lunedì al Venerdì

```
  reg [7:0] Y;
  // Implementazione
  //  A. Con l'istruzione if
  always @(A)
  begin
    if       (A==0) Y=8'b00000001;
    else if (A==1) Y=8'b00000010;
    else if (A==2) Y=8'b00000100;
    else if (A==3) Y=8'b00001000;
    else if (A==4) Y=8'b00010000;
    else if (A==5) Y=8'b00100000;
    else if (A==6) Y=8'b01000000;
    else if (A==7) Y=8'b10000000;
    else           Y=8'b00000000;
  end
  //  B. Con l'istruzione casex
  always @(A)
    begin
      casex(A)
        0: Y=8'b00000001;
        1: Y=8'b00000010;
        2: Y=8'b00000100;
        3: Y=8'b00001000;
        4: Y=8'b00010000;
        5: Y=8'b00100000;
        6: Y=8'b01000000;
        7: Y=8'b10000000;
      default: Y=8'bX;
      endcase
    end
  //  C. Con il loop for
  integer N;
  always @(A)
    begin
      for(N=0; N<=7; N = N + 1)
        if(A == N)
          Y[N] = 1;
        else
          Y[N] = 0;
    end
endmodule
```

MODULO 6
TIPOLOGIE DI OPERATORI VERILOG

A questo punto è opportuno aprire una breve parentesi sugli *operatori* Verilog.
Gli operatori identificano l'operazione da eseguire sui loro operandi per produrre un nuovo valore.
Possono essere:

- binari, se applicati a due operandi
- unari, se si riferiscono a un solo operando

Due eccezioni sono rappresentate dagli operatori *condizionali*, che agiscono su 3 operandi, e quelli di *concatenazione*, che possono avere un numero qualsiasi di operandi. La tabella 3 riassume i vari tipi di operatori Verilog.

Tipo operatore	Simbolo	Funzione svolta
Aritmetici	+ - * / %	Aritmetica Modulo
Relazionali	> >= < <=	Relazione
Uguaglianza	== !=	Uguaglianza logica Disuguaglianza logica
Logici	! && ||	NOT logico AND logico OR logico
Su Bit	~ & | ^ ^~ ~^	Negazione AND OR EXOR EXNOR
Riduzione	& | ~& ~| ^ ^~ ~^	AND OR NAND NOR EXOR EXNOR
Shift	<< >>	Shift a sinistra Shift a destra
Condizionali	? :	Condizione
Concatenazione	{ }	Concatenazione

Tabella 3 *operatori Verilog*

Operatori aritmetici

Effettuano funzioni aritmetiche sugli operandi. Sono classificati in:

- Addizione (+)
- Sottrazione (–)
- Moltiplicazione (*)
- Divisione (/)
- Modulo (%)

Si possono usare gli operatori "+", "–", e "*" con le variabili o le costanti. Gli operatori "+" e "–" sono sia unari che binari. Il compilatore HDL richiede che gli operatori "/" e "%" si applichi-

no a operandi costanti.
Si può osservare che le operazioni di moltiplicazione e divisione sono realizzabili in maniera efficiente solo su quei dispositivi che dispongono delle risorse adeguate per realizzarli. Ad esempio, le FPGA della serie Virtex, per le quali sono disponibili dei moduli moltiplicatori embedded a 18 bit. Per le divisioni, il metodo più efficiente consiste nell'utilizzare dei moduli ad hoc con un applicativo apposito di Xilinx (Coregen).
Esempio di addizione:

```
parameter size=8;
wire [3:0] a,b,c,d,e;
assign c = size + 2; //costante + costante
assign d = a + 1;    //variabile + costante
assign e = a + b;    //variabile + variabile
```

Operatori relazionali

Servono a confrontare tra loro due quantità, dando come risultato "1" o "0" ("1"=vero, "0"=falso).
Gli operatori di relazione in Verilog sono:

- Minore di (<)
- Minore o uguale (<=)
- Maggiore di (>)
- Maggiore o uguale (>=)

Come esempio, cogliamo l'occasione per introdurre una ***funzione*** Verilog che restituisce il massimo tra due valori a 8 bit:

```
function [7:0] max( a, b );
  // Ingressi
  input [7:0] a,b;
  // Impl.
  if ( a >= b ) max = a;
  else max = b;
endfunction
```

Operatori di uguaglianza

Gli operatori di uguaglianza sono i seguenti:

- Uguaglianza (==)
- Disuguaglianza (!=)

Gli operatori di uguaglianza danno come risultato uno "0" logico se gli operandi da confrontare sono diversi, "1" se sono uguali (intendendo il confronto bit a bit).
Come esempio, citiamo il caso di un riconoscitore di codice operativo per un'istruzione di tipo *jump*:

```
module is_jump_instruction (istruzione, jump);
  // Costanti
  parameter JMP = 2'h3;
  // I/O
  input [7:0] instruzione;
  output jump;
  // Impl.
  assign jump = (instruzione[7:6] == JMP);
endmodule
```

È importante sottolineare che solo due dati alla volta possono essere confrontati. Così, un'istruzione come `If ( A = B = C)` non si può usare.
Gli operatori logici, invece, si possono impiegare per testare il risultato di confronti multipli, come in: `If ((A=B) && (A=C))`

Operatori logici

Gli operatori logici restituiscono il valore "1" o "0" a seconda che l'espressione da valutare sia vera o falsa e sono i seguenti:

- NOT (!)
- AND (&&)
- OR (||)

In particolare, l'operatore logico NOT produce "1" se l'operando ha valore "0".

Esempio: si deve confrontare una variabile di ingresso (*dato*) con un certo valore o con il suo negato, in corrispondenza di un criterio di abilitazione (*enable*). In uscita, si genera un segnale di validità (*test_ok*):

```
module is_valid(dato,enable,test_ok);
  input [7:0] dato;
  input enable;
  output test_ok;
  assign test_ok = ((dato == 8'b10101010)
  && (enable == 1'b1) || (dato == 8'b01010101)
  && (enable == 1'b0));
endmodule
```

Operatori su bit

Questi operatori, tradotti con *Bitwise Operators*, sono caratterizzati dall'agire sull'operando bit a bit, tipicamente sui bit di un vettore. Sono i seguenti:

- Negazione unaria (~)
- AND binario (&)
- OR binario (|)
- EXOR binario (^)
- EXNOR binario (^~ or ~^)

Operatori di riduzione

Gli operatori di riduzione Verilog agiscono su un operando e danno come risultato un singolo bit. Sono i seguenti:

- AND (&)
- OR (|)
- NAND (~&)
- NOR (~|)
- EXOR (^)
- EXNOR (^~ or ~^)

Come esempio, presentiamo un modulo che riceve in ingresso un numero binario a 8 bit (*valore*) e restituisce in uscita la sua parità più un criterio che vale *true* se il numero è composto di cifre tutte a "1":

```
module check_input ( valore, parita, tutti_uni );
   // I/O
   input [7:0] valore;
   output parita, tutti_uni;
   // Impl.
   assign parita = ^ valore;
   assign tutti_uni = & valore;
endmodule
```

Operatori di shift

L'operatore di shift riceve due operandi e shifta il valore del primo operando verso destra (o sinistra) per il numero di volte specificato dal secondo operando.
Gli operatori di shift Verilog sono i seguenti:

- Shift vs sinistra (<<)
- Shift vs destra (>>)

Dopo lo shift, i bit rimasti "scoperti" sono riempiti con zeri.
Nell'esempio che segue utilizziamo l'operatore di shift a destra per realizzare una divisione per 4:

```
module divido_per_4( dividendo, quoziente );
   // I/O
   input [7:0] dividendo;
   output [7:0] quoziente;
   // Implementazione
   assign quoziente = dividendo >> 2;
   //shift vs dxt 2 bit
endmodule
```

Operatore condizionale

L'operatore condizionale (? :) valuta un'espressione e restituisce il valore *vero* ("1") o *falso* ("0").
La sintassi dell'istruzione di *assegnazione condizionale* è la seguente:

```
assign <SEGNALE> = <CONDIZIONE> ?
<VALORE_1> : <VALORE_2>
```

dove si intende che alla variabile <SEGNALE> viene assegnato il <VALORE_1> se la <CONDIZIONE> è vera, oppure il <VALORE_2> se è falsa.

Esempio: dati due ingressi numerici (*a e b*) e un codice operativo (*codice*), si valuta se l'espressione (codice == ADD) è vera. In caso affermativo, all'uscita (*risultato*) si assegna il valore *a* + *b*, altrimenti il valore *a* – *b*.

```
module add_or_sub( a, b, codice, risultato );
   // Costanti
   parameter ADD=1'b0;
   // I/O
   input [7:0] a, b;
   input codice;
   output [7:0] risultato;
   // Implem.
   assign risultato = (codice == ADD) ?
    a+b : a-b;
endmodule
```

Caso particolare: si possono anche "nestare" ("annidare") più operatori condizionali per produrre un costrutto di tipo if ... else. L'esempio seguente mostra come impiegare l'operatore condizionale per riconoscere un codice operati-

ELETTRONICA EXPO DAY

MOSTRA MERCATO

NOVI LIGURE

CENTRO FIERISTICO "Dolci Terre di Novi" VIALE CAMPIONISSIMI

DOMENICA 8 GENNAIO 2006

dalle ore 9 alle ore 18.30

ELETTRONICA - HARDWARE - TELEFONIA
COMPONENTISTICA - SURPLUS - MATERIALE DI CONSUMO
RADIANTISMO - ACCESSORI - RICEZIONE SATELLITARE

SI POTRÀ TROVARE DI TUTTO...DALLE ANTICHE RADIO IN BACHELITE ALLE NUOVISSIME CONNESSIONI WIRELESS!!

Organizzazione
EXPOFIERE / EXPOEVENTI

INFO LINE: 333.6252325
FAX 0533.8531117

E-mail: expo-fiere@libero.it
www.expofiere.com

vo ed eseguire la corretta operazione:

```
module aritmetico( a, b, codiceop, risultato );
 // Costanti
 parameter  ADD=3'h0, SUB=3'h1, AND=3'h2,
 OR=3'h3,  XOR=3'h4;
 // I/O
 input [7:0] a,b;
 input [2:0] codiceop;
 output [7:0] risultato;
 // Implem.
 assign risultato = ((codiceop == ADD) ? a+b : (
                    (codiceop == SUB) ? a-b : (
                    (codiceop == AND) ? a&b : (
                    (codiceop == OR ) ? a|b : (
                    (codiceop == XOR) ? a^b :
                    (a)))))));
endmodule
```

Operatori di concatenazione

La concatenazione combina uno o più operandi per formare un vettore più ampio.
In Verilog, si indica mettendo in una lista gli operandi da concatenare, separati da virgole e il tutto si racchiude tra parentesi graffe. Ad esempio, concatenare {1'b1,1'b0,1'b0} dà come risulato 3'b100.
Si può anche, ove possibile, raccogliere i termini a "fattor comune", come nella più classica notazione matematica. Come caso tipico, l'espressione `{1'b1,1'b0,1'b0}` equivale alla `{1'b1,{2{1'b0}}}`

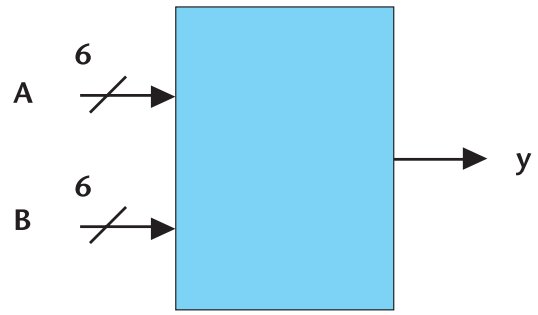


Figura 9 Schema a blocchi del comparatore

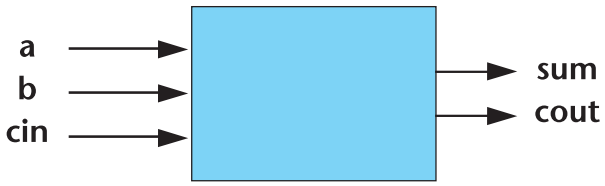


Figura 10 Full adder a 1 bit

ESEMPIO 5: COMPARATORE

Il codice sorgente per un semplice comparatore a 6 bit è riportato di seguito. I comparatori si possono modellare convenientemente con l'istruzione *if ... else*.

Comparatore a 6 bit

Questo modulo riceve in ingresso due valori binari a 6 bit, li confronta bit a bit ed emette un'uscita a livello "1" se sono uguali, "0" se trova almeno una coppia di bit diversi.

```
module compar_6 (A,B,Y);
  // I/O
  input [5:0] A, B;
  output Y;
  integer N;
  reg Y;
  // Impl.
  always@(A or B)
  begin
    Y = 1;
    for (N=0; N<6; N=N+1)
      if(A[N] != B[N])
        Y=0;
  end
endmodule
```

ESEMPIO 6: SOMMATORE (ADDER)

In figura 10 è riportato lo schema a blocchi di un *full-adder*. Si tratta di un sommatore a 1 bit che riceve in ingresso due operandi (*a* e *b*) più un riporto (*carry_in*) e restituisce la somma (*a* + *b*) più il riporto di uscita, indicato con *carry_out*. La somma può essere facilmente estesa a più bit, come vedremo.
Fa uso degli operatori di *riduzione* ^ (EXOR) e & (AND).

```
module full_adder (a,b,cin,sum,cout) ;
  // I/O
  input a,b,cin ;
  output sum,cout ;
  wire s ;
  // Impl.
  assign s = a ^ b ;
```

```
  assign sum = s ^ cin ;
  assign cout = (a & b) | (s & cin) ;
endmodule
```

Per sommatori più ampi possiamo *parametrizzare* il codice, come nell'esempio seguente che fa uso della dichiarazione *parameter*.
Con essa si definisce la costante "size" pari a 8. L'adder è a 8 bit:

```
module adder_8 (sum, a, b) ;
  // Cost.
  parameter size = 8 ;
  // I/O
  input [size - 1:0] a ;
  input [size - 1:0] b ;
  output [size - 1:0] sum ;
  // Impl.
  assign sum = a + b ;
endmodule
```

Il tool di sviluppo ISE Webpack non ha difficoltà a riconoscere questa codifica e ad *inferenziare* un blocco sommatore da 8 bit, come confermato dal file di report del processo di Sintesi e dal simbolo di schematica che viene prodotto (figura 11).

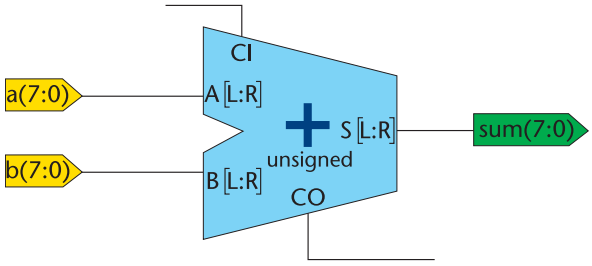


Figura 11 Simbolo grafico del blocco adder a 8 bit

PROSSIMAMENTE

Nella prossima puntata tratteremo l'interfaccia per la programmazione in-system di CPLD Xilinx e analizzeremo altri elementi fondamentali della sintassi Verilog. Saranno presentati, inoltre, anche diversi esempi di codice.

S.V.M. ELETTRONICA
Vendita per corrispondenza Tel./Fax 0331/640569
Caratteristiche tecniche e vendita on-line **www.svmelettronica.com**

INVERTER 12VDC/220VAC SOFT-START
WHS150W-12 €. 40,00
WHS200W-12 €. 45,00
WHS300W-12 €. 52,00
WHS400W-12 €. 60,00
WHS600W-12 €. 95,00

Sistema di videosorveglianza wireless operante sulla banda dei 2,4GHz composto da 4 telecamere a colori da esterno con illuminatore IR a 12 LED, e da un ricevitore multicanale con switcher e telecomando.
COD. 37/310 €. 360.00

Navigatore Mio268 Sw Italia
SD 256Mb SW Italiano.
COD. RC268I €. 422,00

Kit telecamera 1/3CMOS waterproof a colori + ricevitore senza fili con illuminatore IR operante sulla banda dei 2.4GHz.il kit include telec., ricev. a 4 canali e alimentatori.
COD. 37/150 €. 115,00

Multimetro digitale
con ingressi protetti
COD. TE/6300E €. 48,00

Tutti i prezzi si intendono IVA inclusa.

Decima parte
n° 244 - Ottobre 2005
Switching flyback multi-uscita

Undicesima parte
n° 245 - Novembre 2005
Analisi di uno switching off-line (parte I)

Dodicesima parte
n° 246 - Dicembre 2005
Analisi di uno switching off-line (parte II)

Alimentatori switching:

Siamo arrivati alla conclusione del nostro lungo, ma spero interessante, percorso. In questa ultima puntata completeremo l'analisi dell'alimentatore switching da PC, con la presentazione di alcuni fondamentali stadi: quello di potenza, quello di uscita, l'alimentatore ausiliario ed il PFC. Buona lettura!

INTRODUZIONE

Nella scorsa puntata avevamo iniziato ad esaminare un alimentatore ATX per PC, ed a ricostruirne lo schema elettrico. Vediamo a che punto eravamo arrivati, con lo schema elettrico dello stadio di ingresso riportato in figura 1. I più attenti avranno notato la presenza di un'ulteriore connessione, evidenziata in blu in figura 1, e precisamente la connessione del punto di mezzo dei condensatori di filtro, denominata +HV/2 (circa 162,5V), che sarà utile nella discussione che segue. All'uscita di questo stadio avremo una tensione continua massima di circa 325V, con tolleranza +-15%: raccomando ancora una volta la massima prudenza nel caso vogliate sperimentare su simili alimentatori!

L'ALIMENTATORE AUSILIARIO

Dalla puntata numero 10, pubblicata su FE 244 (Ottobre 2005), sappiamo che il circuito integrato PWM ha bisogno di una tensione relativamente bassa per funzionare, tipicamente 10-20V, e che questa tensione viene ottenuta per mezzo di configurazioni molto semplici e funzionali.
Nel nostro caso, a complicare le cose, l'alimentatore ATX deve **anche** fornire una tensione di +5V, denominata di Stand-by, necessaria alla scheda madre per l'implementazione di funzionalità tipo Wake-On-Lan e, più banalmente, per poter permettere l'accensione tramite un pulsante (non è un interruttore come nei vecchi PC) e lo spegnimento via software.
La specifica ATX V2.1 consiglia, ma non obbliga, una capacità di corrente pari a 2A per i +5VSB, con una tolleranza di +-5% sulla tensione di uscita. L'etichetta del nostro alimentatore PC (FE 245) riporta invece un massimo di 1,5A, quindi inferiore a quanto consigliato dalla specifica: ciò non vuol dire che il nostro alimentatore non funzioni correttamente con la stragrande maggioranza dei PC, ma solo che si ottiene quello che si paga! Lo schema completo dell'alimentatore ausiliario è riportato in figura 2. Per riferimento, in figura 3 è visibile il circuito reale dell'alimentatore ausiliario (della sola parte a sinistra del trasformatore T3), di dimensioni pari a circa 20x45mm in pianta.

Procediamo ora con la descrizione tecnica. L'alimentatore in questione è in tipologia flyback (vi ricordate che questa è una tipologia molto usata, per potenze inferiori a 150W circa?), e presenta, come al solito, alcune soluzioni circuitali tese a ridurre al minimo il numero di componenti. Vi chiederete: ma dov'è l'oscillatore? ed il circuito di controllo PWM? E l'amplificatore di errore? Continuate a leggere...
La resistenza R15 fornisce una piccola corrente di base al transistor oscillatore Q3. Questa corrente si può valutare applicando la legge di Ohm:

$$I_{R15} = (HV - VBE_{Q3}) / R15 = (325 - 0{,}7) / 220000 = 1{,}47mA$$

Di questa corrente, una parte scorre attraverso la serie di R16 e R17, esattamente:

$$I_{R18} = VBE_{Q3} / (R16 + R17) = 0{,}7 / (1000 + 10) = 0{,}69mA$$

Analisi di uno switching off-line (parte II)

di Romano Bernarducci
r.bernarducci@farelettronica.com

Resta quindi disponibile, per la corrente di base di Q3:

$$IB_{Q3} = 1,47 - 0,69 = 0,78mA$$

in quanto è immediato constatare che nessuna corrente scorre inizialmente attraverso ZD1. Per la precisione una piccola corrente va a caricare il condensatore C12 alla tensione di 0,7V circa, ma possiamo ignorarla. La corrente di base attiva debolmente il transistor Q3, del tipo KSC5027-R della Fairchild Semiconductor. Dal data-sheet, facilmente reperibile sul web all'indirizzo *www.fairchildsemi.com/pf/KS/KSC5027.html*, notiamo trattasi di un transistor NPN per alta tensione (fino a 800V tra Collettore ed Emettitore), di media corrente (3A), e con un beta abbastanza basso, compreso tra 15 e 30 (vedi riquadro per i neofiti).

Il h_{FE} di un transistor

Rappresenta il coefficiente di amplificazione in corrente del transistor, cioè il numero, che moltiplicato per la corrente di BASE, fornisce la corrente di COLLETTORE.
ATTENZIONE: utilizzare per la progettazione, un preciso valore del beta, desunto dal data-sheet o misurato con un tester, è spesso fuorviante e causa di errori, in quanto esso varia moltissimo da esemplare ad esemplare, ed al variare delle condizioni operative (polarizzazione, temperatura, tipo di segnale AC/DC).

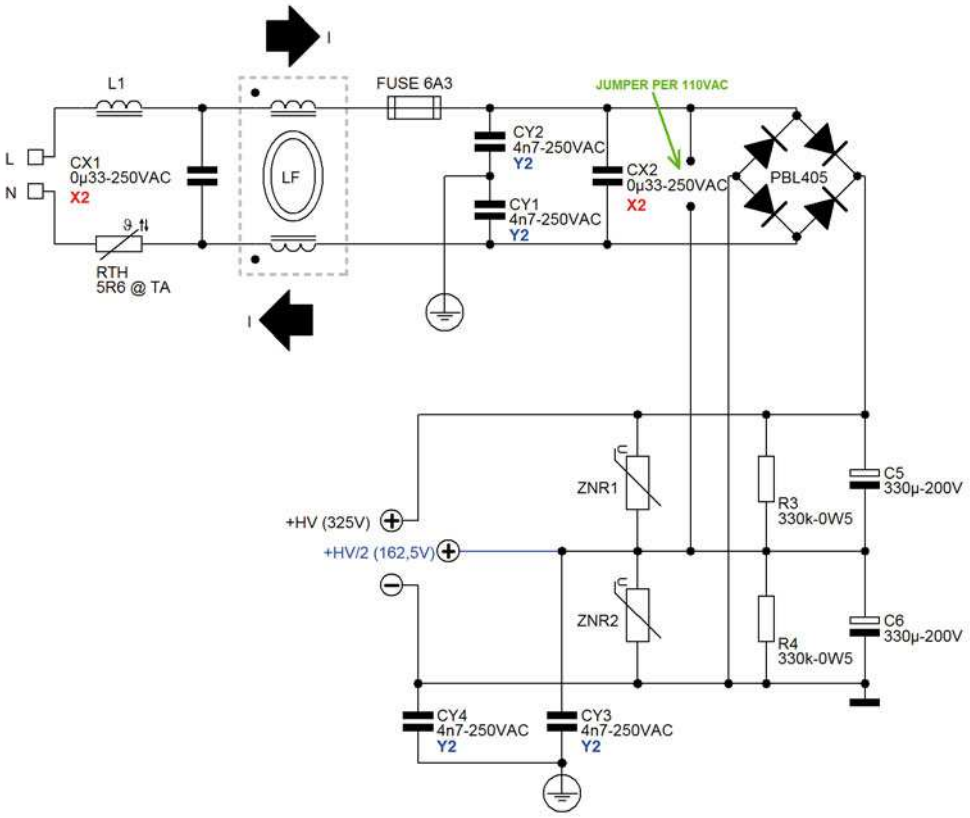


Figura 1 Schema dello stadio di ingresso (senza filtro

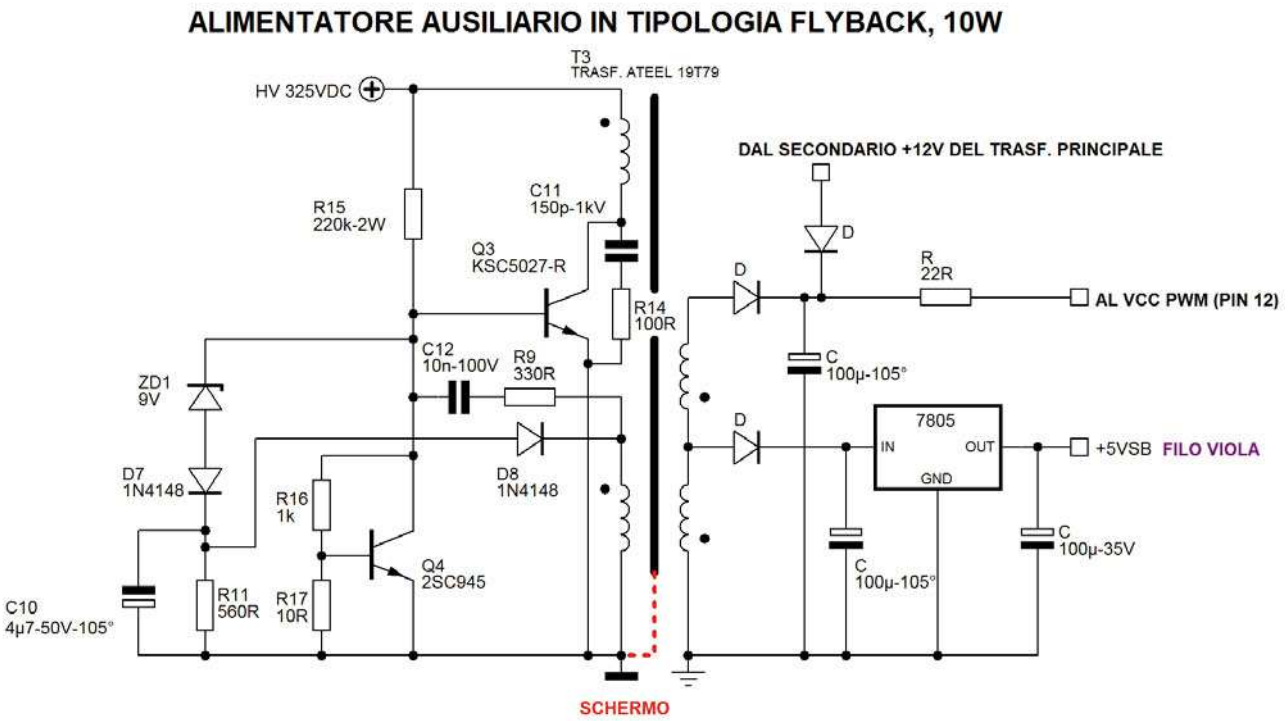


Figura 2 Schema elettrico dell'alimentatore ausiliario

L'attivazione di Q3 collegherà a massa il terminale inferiore del primario del trasformatore T3, che quindi vedrà ai suoi capi l'intera tensione di alimentazione (325V). A questo punto, ai capi dell'avvolgimento in basso a sinistra del trasformatore T3, comparirà istantaneamente una tensione pari a 325V diviso per il rapporto spire tra i due avvolgimenti primari, secondo le note regole dei trasformatori.
Dai "puntini" degli avvolgimenti, questa tensione avrà polarità concorde con quella dell'avvolgimento primario (in alto a sinistra del trasformatore), e attraverso il condensatore C12 e la resistenza R9, farà scorrere una corrente supplementare sulla base di Q3, **aumentandone sensibilmente la velocità di commutazione.**
In pratica si ha una reazione positiva! Lo scopo del condensatore C12 è quello di limitare il tempo di intervento della reazione positiva, infatti la corrente che vi scorre, lo carica gradualmente, quindi si riduce, ed infine si annulla a carica completata.
La costante di tempo, indicata con la lettera greca τ (tau), è data dal semplice prodotto R*C, e vale in questo caso:

$$\tau = R9 * C12 = 330 * 10e\text{-}9 = 3300ns = 3{,}3\mu s$$

Dalle leggi dell'elettrotecnica sappiamo che dopo circa 3 * τ, cioè dopo circa 10µs, il condensatore sarà carico e la corrente aggiuntiva andrà a zero.
Come si realizza l'oscillazione? Abbiamo appena visto che, alla commutazione ON di Q3, la sua corrente di base aumenterà di colpo, per poi scendere ed annullarsi dopo circa 10µs. Nel medesimo periodo la corrente di collettore aumenterà linearmente.

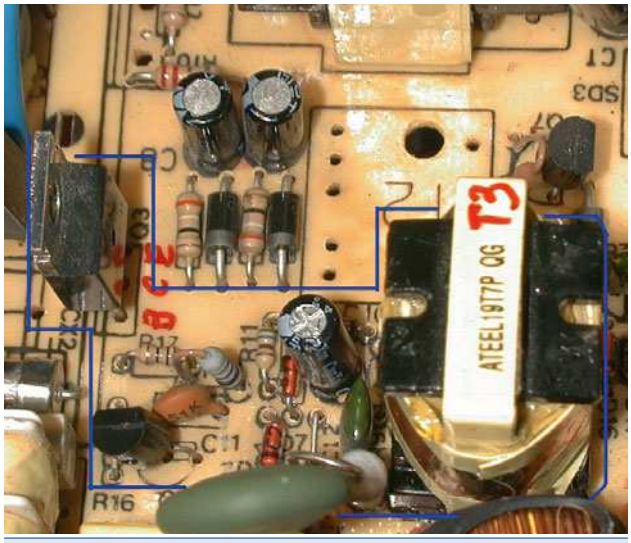


Figura 3 L'alimentatore ausiliario

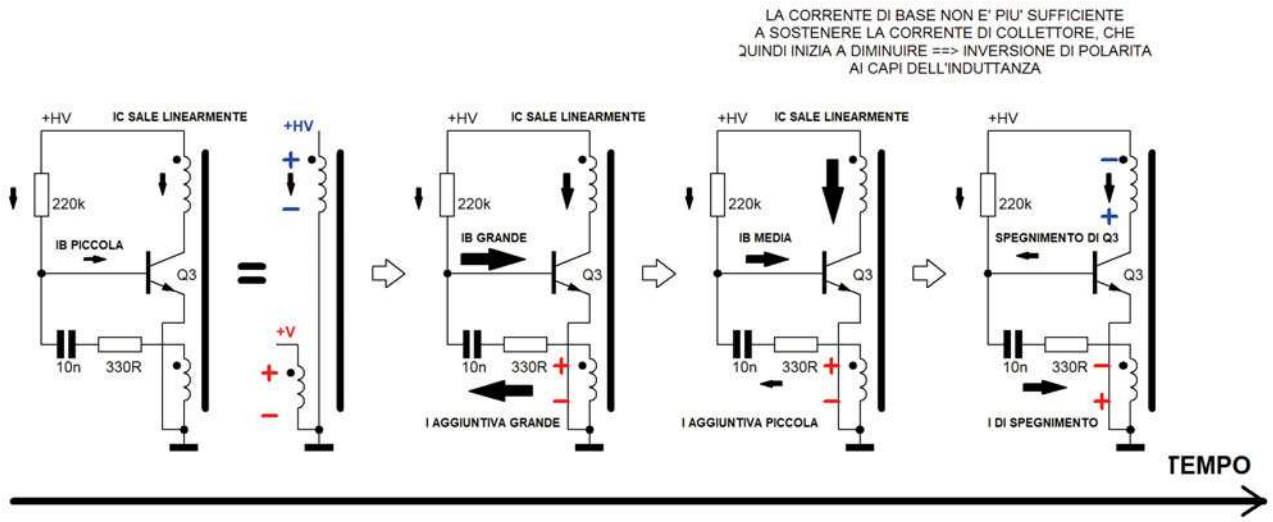


Figura 4 *Ciclo di funzionamento del flyback a reazione positiva*

Ma un transistor "ON" (in saturazione) riesce a far scorrere una certa corrente di collettore finché la corrente di base è sufficiente, secondo il fattore moltiplicativo dato dal beta.

Non appena la corrente di collettore **(in crescita)**, supererà di **h_{FE} volte** la corrente di base **(in diminuzione), la corrente di collettore inizierà a diminuire.** Ma questa è la stessa corrente che scorre nell'avvolgimento del primario (cioè un'induttanza), ed è noto che **una riduzione della corrente è possibile solo in concomitanza dell'inversione della tensione ai capi dell'induttanza stessa!** L'inversione di tensione si rifletterà istantaneamente anche sull'avvolgimento in basso, aiutando a spegnere rapidamente il transistor. Per i motivi appena citati l'avvolgimento in basso sul primario viene spesso denominato **"avvolgimento di feedback"**.
La figura 4 prova a fare un poco di ordine. Le frecce nere indicano la direzione della corrente e sono più o meno grandi a seconda dell'intensità della corrente.

Ok, abbiamo risolto il problema dell'oscillazione, come fare adesso per la regolazione della tensione di uscita? Semplice...
L'avvolgimento di feedback, **durante la fase OFF**, si comporta anche come un classico secondario flyback, e carica il condensatore C10 ad una tensione negativa, attraverso il diodo D8 (infatti in questa fase la tensione sull'avvolgimento di feedback è invertita rispetto alla massa).
La tensione negativa livellata da C10, regola la commutazione di Q3. Infatti se la tensione negativa su C10 salisse troppo in valore assoluto, ad esempio passando da -10V a -11V, lo zener composto da ZD1 e D7 entrerebbe in conduzione, abbassando la tensione di base di Q3 ad un valore sufficiente a spegnerlo. Viceversa, se la tensione su C10 fosse inferiore al valore desiderato, ad esempio -7V, i diodi ZD1 e D7 non condurrebbero, e Q3 sarebbe libero di oscillare alla sua massima potenza.
A questo punto, per le note proprietà dei flyback multiuscita, anche i veri secondari, alla destra del trasformatore T3 di figura 2, saranno regolati ad un valore di tensione pari al rapporto tra il numero delle proprie spire e quelle dell'avvolgimento di feedback.
E adesso non ditemi che questa non è una applicazione veramente intelligente!!

Quesito

Qual'è la funzione del transistor Q4, con la rete di polarizzazione costituita da R16 e R17? Sembra un classico circuito denominato *"VBE multiplier"* o *"active diode"*.
Inviatemi le vostre idee...

La resistenza R14 ed il condensatore C11 costi-

tuiscono la cosiddetta rete *"snubber"*, cioè hanno lo scopo di ridurre gli elevatissimi spikes di tensione che si generano alla fine della fase ON, sul collettore di Q3. Per finire, sul secondario del trasformatore, notate i due avvolgimenti che forniscono circa +11V al circuito integrato PWM, e circa +8V per la VSB. Questi ultimi vengono ridotti a +5V da un banalissimo regolatore lineare di tipo 7805. Questo spiega perchè la +5VSB è limitata a 1,5A... è infatti questo il limite per i regolatori lineari in contenitore TO220. Anzi, in generale i comuni 7805 sono forniti per una corrente massima di 1A, e solo alcuni "reggono" fino a 1,5A. Per quanto riguarda il circuito di controllo PWM (un classico TL494), notate come, in stand-by, sia alimentato dall'alimentatore ausiliario. Una volta acceso lo switching di potenza principale, subentra a quello ausiliario nel fornire l'alimentazione al controller PWM, tramite il diodo in verticale, in alto a destra, in figura 2; si evita così di caricare inutilmente il piccolo flyback ausiliario.

IL TRASFORMATORE

Il trasformatore switching per l'alimentatore ausiliario, è costruito su un nucleo in ferrite di dimensioni 28x20mm, spessore 5mm, visibile in figura 5. Non vi sembra un po' grande, per soli 10W circa di potenza fornita? Se ricordate (FE di giugno 2005) un più piccolo nucleo di tipo E20/10/6 poteva in teoria trasferire 26W a 25kHz e ben 118W a 100kHz, come mai questo sovradimensionamento?
La risposta è semplice: in un flyback, la tensione sul collettore del transistor, durante il periodo OFF, può arrivare al doppio della tensione di alimentazione, più una sovratensione che dipende dalla realizzazione del trasformatore (dipende dalla cosiddetta **induttanza di perdita**, in ingle-

Figura 5 Il trasformatore per lo switching ausiliario

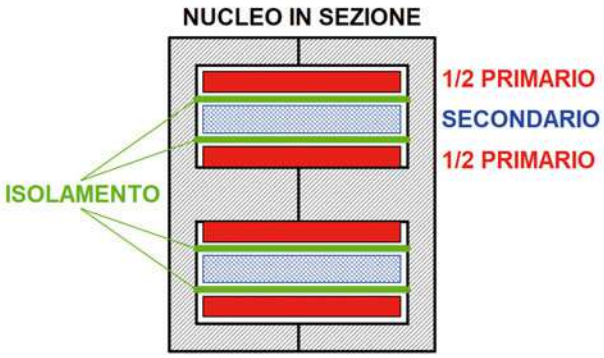


Figura 6 Avvolgimento del secondario in mezzo a due semi-primari

L'isolamento nei trasformatori per switching off-line

1. I trasformatori switching per alimentatori off-line, devono sottostare a particolari e severe normative riguardo alle distanze minime di isolamento tra gli avvolgimenti.
2. Le distanze minime di isolamento possono essere anche di diversi millimetri.
3. I trasformatori con piccolo nucleo sono maggiormente penalizzati, in quanto una consistente percentuale del loro nucleo NON può essere utilizzata per gli avvolgimenti.

Ne conseguono due mie piccole personale considerazioni, relative agli alimentatori off-line:

1. Per piccole potenze, uguali o inferiori a 2-3W, è preferibile utilizzare un alimentatore lineare. Un trasformatore a 50Hz da 1,2VA ha dimensioni in pianta di soli 32x27mm.
2. Realizzare in casa un trasformatore switching per applicazioni off-line richiede ESPERIENZA. Esercitatevi preliminarmente con trasformatori switching per bassa tensione.

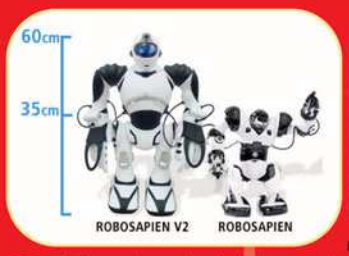
se **"leakage inductance"**, la cui trattazione esula però dagli scopi di questo corso base), e quindi può valere circa 700V-800V. Questa medesima tensione, sarà presente anche sul terminale inferiore del primario del trasformatore. Il problema è che **il secondario del trasformatore, che DEVE essere ben isolato dal primario in quanto può venire in contatto con l'utente**, viene di norma avvolto direttamente sopra al primario, ed in alcuni casi addirittura in mezzo a due semi-primari,come in figura 6.
Esistono stringenti e severe normative, come la IEC/EN 60950, che regolano la **minima** distanza che deve essere garantita tra il primario ad alta tensione ed un qualsiasi secondario in bassa tensione; un sito web interessante che riporta un software on-line per il calcolo delle cosiddette distanze di **creepage** e di **clearance**, in accordo con la normativa citata, è *www.creepage.com.* Riassumiamo nel riquadro questa ed altre considerazioni. Per concludere l'analisi del trasformatore, notate dalla figura 5, la bandella di rame che realizza una spira di cortocircuito collegata a massa dal filo nudo. **Attenzione, la spira non passa all'interno del trasformatore, ma all'esterno, e il suo scopo è quello di ridurre le emissioni elettromagnetiche.** La sua presenza è comune nei trasformatori per flyback, dove le maggiori emissioni sono proprio causate dal piccolo gap di aria presente sulla gamba centrale del nucleo.

GLI STADI DI POTENZA (COMMUTAZIONE ED USCITA)

In figura 7 è riportato lo schema elettrico della stadio di commutazione, ed una versione semplificata dello stadio di uscita. Questo stadio di potenza è in una tipologia per la quale non abbiamo mai presentato le formule di progetto, in quanto di difficile realizzazione per l'hobbysta medio. Si tratta della tipologia **half-bridge**, cioè **mezzo-ponte**.

Come tutte le tipologie, anche questa presenta vantaggi e svantaggi, cominciamo da questi ultimi:

IDEA ELETTRONICA: accendiamo le tue idee!



ROBOSAPIEN 2

Il giocattolo si evolve parla, canta, balla, gioca, raccoglie e lancia oggetti, e fa persino mosse di kung fu. Si chiama Robosapien V2, e rappresenta la nuova generazione di robot per il divertimento domestico, presentati in anteprima al CES di Las Vegas. Dotato di videocamere e radar a infrarossi, il robottino alto 60 cm è in grado anche di riconoscere le persone, interagendo con loro.

€ 299,00

MINI ROBOSAPIEN - edizione limitata -

Fratello più piccolo del Robosapien! Questo piccolo robot ha funzionalità limitate rispetto al grande Robosapien, ma è fantastico per gli appassionati di Robosapien. Misura appena 18 centimetri di altezza e può camminare. I suoi caratteristici occhi rossi si illumineranno mentre cammina. Le pinze di presa funzionano con i tasti meccanici ed i relativi bracci possono essere spostati su e giù usando gli scorrevoli sulla parte posteriore delle sue spalle. Il mini Robosapien utilizza due batterie AAA .

BIANCO € 20,00 - ARGENTO/BLU/ORO € 23,00

ROBOSAPIEN COLORATI

- edizione limitata -

Robotsapien è un mini Robot Androide progettato da Mark Tilden, creatore dei Robots B.e.a.m. e degli stupefacenti B.I.O. Bugs. Robosapien compie Movimenti e gesti fluidi: camminata veloce e dinamica a due velocità; braccia completamente funzionali con due tipi di presa delle mani, 67 funzioni pre-programmate: presa, lancio, calcio, danza, kung-fu, aerofagia, eruttazione, rap e molte altre ancora, 4 modalità di programmazione, 3 modalità dimostrative.

€ 125,00

MINI ROBOT PICAXE

Il Microrobot Picaxe consiste in una base in plastica di 120 X 80mm che include 2 motoriduttori (rapporto 42:1) con relative ruote in gomma e ospita 4 batterie stilo, una scheda che e' composta da un PICAXE 18 (espandibile con un PICAXE18X) un chip che genera automaticamente gli impulsi PWM per comandare i motori e un chip di potenza che pilota i due motori (L293D). La funzione PWM e' molto interessante in quanto permette di modulare la velocita' dei motori in modo molto graduale, senza impegnare il PICAXE, lasciando quindi spazio al programma di gestione dei sensori e della logica del Robot. Sono disponibili ben 5 Input per collegare vari sensori e 4 Output. La scheda e', inoltre, provvista del comodo connettore da 3.5mm stereo, per la programmazione seriale, in questo modo tutte le operazioni di programmazione e debug avvengono in modo rapido e sicuro.

€ 84,00

Tutti i prezzi si intendono IVA compresa. Per ordini e informazioni:
IDEA ELETTRONICA - Via San Vittore n°24/A - 21040 Oggiona con S. Stefano - Varese - ITALY - Tel./Fax 0331/212723
Visitate il nostro sito: WWW.IDEAELETTRONICA.IT

- Richiede due dispositivi di commutazione sul primario.
- Richiede due raddrizzatori sul secondario ed un secondario doppio, per ogni tensione di uscita. In alternativa un secondario singolo ed un ponte di diodi per ogni tensione di uscita.

Vantaggi:

- Utilizza sempre tutto l'avvolgimento primario, a differenza del push-pull in cui viene attivata solo metà primario alla volta. Questo si traduce generalmente in un minore ingombro del trasformatore oppure in una maggiore potenza disponibile a parità di trasformatore.
- I transistor di commutazione sul primario devono sopportare solo la tensione raddrizzata di rete, cioè 325V +15% circa, a differenza ad esempio della tipologia flyback, dove il singolo transistor deve sopportare una tensione pari a circa 700-800V.
- É possibile utilizzare lo stesso circuito, sia per i 110VAC che per i 230VAC, tramite un solo ponticello e mantenendo la stessa efficienza.

Così come già nella topologia push-pull, i due transistor Q1 e Q2 vengono pilotati alternativamente, **ma con un buon margine di sicurezza durante il quale entrambi sono spenti.** Infatti dall'esame dello schema è immediato verificare che una conduzione contemporanea di Q1 e Q2, porterebbe ad un cortocircuito secco della tensione di +325V verso massa, con conseguenze esplosive. Per questo motivo viene in genere utilizzato un tempo di OFF (dead-time) pari almeno al 20% del periodo di oscillazione.

Una particolare menzione merita il trasformatore T2, che normalmente ha il compito di pilotare le basi dei due transistor. Ma questo sarebbe stato troppo poco, infatti, T2 presenta un terzo avvolgimento, utilizzato come sensore di corrente per il primario! É veramente un componente bidirezionale: da destra a sinistra (figura 7) trasporta ed isola il segnale di pilotaggio dal controller PWM alle basi dei transistor di potenza, da sinistra a destra trasporta l'informazione sulla corrente che scorre nel primario.

Per motivi che sarebbe lungo spiegare, il controllo di corrente ha lo scopo di evitare la saturazione del nucleo, con le solite e note nefaste conseguenze.

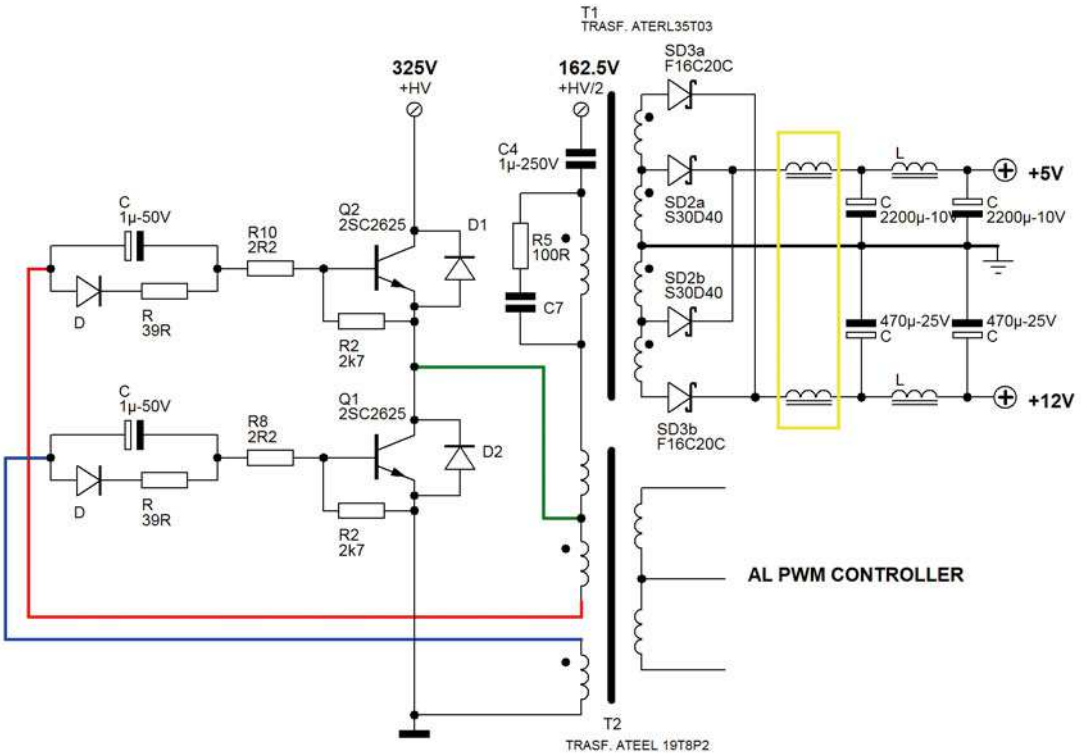


Figura 7 Schema elettrico della stadio di commutazione e di uscita (semplificato

Notate tuttavia che non è un vero sistema di controllo Current-Mode, in quanto il controller PWM utilizzato (TL494) è del tipo Voltage-Mode.

Dalla figura 7, quando il "pallino" di T2 è positivo rispetto all'altro capo dell'avvolgimento, Q1 è acceso (filo blu positivo sulla Base, Emettitore a massa) mentre Q2 è spento. Viceversa quando il "pallino" di T2 è negativo, Q1 è spento mentre Q2 è acceso (filo rosso positivo sulla Base, filo verde negativo sull'Emettitore).
Notate che nello stadio di uscita, due induttanze sono racchiuse entro un riquadro di colore giallo: questo indica un unico nucleo, nel caso specifico toroidale, su cui sono avvolte tutte le induttanze allo scopo di migliorare la cross-regolazione delle varie uscite.
Dallo schema ho omesso le uscite per le tensioni di -5V e -12V, che si ottengono banalmente dagli stessi secondari (cioè il -5V dai due secondari del +5V, il -12V dai due secondari del +12V), utilizzando diodi e condensatori con **polarità invertita** rispetto a quelli mostrati in figura.
Dallo schema manca anche l'uscita ad alta corrente dei +3,3V. Questa tensione si ottiene a partire dai due secondari a +5V, per mezzo di un interessante regolatore denominato *"amplificatore magnetico"*, costituito da un avvolgimento su una ferrite speciale (e' una ferrite che presenta un ciclo magnetico B-H quasi perfettamente rettangolare), che si comporta come **un interruttore comandato a bassissima perdita.**
Questa trattazione esula però dai limiti del corso. Per concludere il paragrafo, in figura 8 notate i tre dispositivi di commutazione (a sinistra il transistor del flyback ausiliario), montati sulla minuscola aletta di 58x67mm: ricordate che questo è possibile solo in virtù del raffreddamento forzato tramite ventola.

CENNI SUL POWER FACTOR CORRECTION

Consideriamo una semplice resistenza: sappiamo che la tensione V ai suoi capi e la corrente I che vi scorre attraverso, sono sempre legate, in qualsiasi istante, dalla Legge di Ohm. Se la tensione V ai capi è costante e continua, la corrente I scorrerà sempre nello stesso verso ed avrà sempre la stessa intensità.
Supponiamo ora che la tensione non sia più costante, ma vari in maniera sinusoidale in ampiezza ed in polarità, sia cioè una **tensione alternata**: è quella che troviamo nelle nostre prese di corrente. Se mettiamo una resistenza ai capi di una presa (non fatelo veramente se non conoscete i rischi connessi!), la corrente che vi scorre sarà anch'essa variabile, sia d'ampiezza che di polarità, ma **ad ogni istante la sua intensità e il suo verso sarà comunque sempre dato dalla Legge di Ohm.**
Per la resistenza, si dice spesso che la forma d'onda della tensione e quella della corrente sono in fase, cioè ancora: al massimo della ten-

Figura 8 I transistor di potenza sul loro dissipatore

Figura 9 Il PFC attivo basato su integrato ST L6561

sione V corrisponde il massimo della corrente I, al minimo della V il minimo della I, e se V = 0, anche I = 0.

Il fattore di potenza, indicato con cosφ, rappresenta il coseno dell'angolo di sfasamento tra tensione e corrente.

Per la resistenza pura, lo sfasamento φ vale 0, e quindi cosφ = cos(0) = 1. Purtroppo le cose cambiano per i componenti cosiddetti reattivi, cioè i condensatori e le induttanze. Per questi, la tensione V e la corrente I non sono più in fase, ma sfasate esattamente di 90°. In pratica quando V=0, I è al massimo (o al minimo), e viceversa. Ne risulta un fattore di potenza cosφ = cos(90°) = 0. Senza addentrarci nella trattazione dettagliata, per la quale rimando ad un buon testo di Elettrotecnica, sappiate che il distributore dell'energia elettrica (Enel in Italia) non gradisce che colleghiamo alla rete elettrica dei carichi con cosφ molto minore di 1. Se non erro è addirittura previsto un sovrapprezzo sul costo dell'energia, nei casi di cosφ basso! A parte il distributore nazionale, sono in vigore comunque le normative Europee IEC/EN 61000-3-2 e 61000-3-4, che specificano, tra le altre cose, i limiti minimi accettabili per il fattore di potenza.

Quale è la ragione di questa premessa?
Avete indovinato?
La risposta è che gli alimentatori switching, hanno un cosφ che lascia abbastanza a desiderare... questo è immediato da comprendere considerando che al loro ingresso ci sono sempre dei condensatori di grande capacità, che vengono caricati impulsivamente dai diodi rettificatori.
Per questa ragione gli alimentatori per PC di una certa potenza (in genere da 300W in su), di recente includono anche un circuito di Power Factor Correction, cioè un circuito in grado di elevare il cosφ a valori più vicini all'unità. I circuiti PFC normalmente utilizzati si dividono in due grandi categorie: **passivi** ed **attivi**. I primi sono costituiti semplicemente da una o due induttanze di elevato valore (decine/centinaia di milliHenries), poste in serie all'alimentazione. Poiché gli sfasamenti del condensatore e dell'induttanza sono opposti, tenderanno a compensarsi e a far risalire il cosφ. La stessa cosa avviene, a parti invertite, nei motori elettrici che in pratica sono assimilabili alle induttanze: un condensatore, appropriatamente chiamato "di rifasamento", riporta il cosφ a valori accettabili. I PFC attivi implementano invece uno speciale switching di tipo **boost**, che converte la tensione in continua, senza però utilizzare grossi condensatori. La tensione viene poi elevata (essendo un boost), al valore costante di circa 400V, **con l'importante accorgimento di prelevare corrente dalla linea elettrica, sempre in fase con la tensione**. I PFC passivi migliorano di molto il cosφ rispetto ad una soluzione senza PFC, ma ovviamente non possono competere con i PFC attivi, i quali riescono tranquillamente ad arrivare a cosφ di 0,98 o più. La figura 9 riporta il PFC attivo presente nell'alimentatore PC, basato sul circuito integrato L6561 della ST, al cui sito *www.st.com* rimando per il data-sheet e le interessanti Application Notes. Da notare infine che le dimensioni del circuito sono solo di 65x65mm, per un altezza di 35mm.

CONCLUSIONI

Trattare un argomento così vasto in 12 puntate, è stato abbastanza difficile. Mi scuserete se in alcuni casi la trattazione generale ha difettato un poco di organicità, ma ogni puntata è stata scritta "all'impronta", senza possibilità di "sistemare" le precedenti puntate, nei casi, abbastanza rari per fortuna, in cui sorgevano delle confusioni od incongruenze. La totalità dei circuiti presentati sono stati realizzati su millefori, quindi vi garantisco il funzionamento a patto di rispettare le poche elementari regole sulle saldature e sulla lunghezza dei collegamenti. Come ultimo consiglio, vi suggerisco di scaricare ed installare sul vostro PC, il software gratuito *SwitcherCad III* dal sito della *Linear Technology*: anche se è molto indirizzato verso l'utilizzo dei componenti della medesima casa, è ottimo comunque, utilizzando i componenti standard (induttanze, capacità, transistor, diodi, ecc.) per provare in pochi minuti la teoria che ho provato a spiegare.

RINGRAZIAMENTI

Ringrazio la Redazione di Fare Elettronica, nella persona di Tiziano Galizia, per la fiducia accordatami durante tutti questi lunghi mesi. Ringrazio poi i lettori per la loro pazienza, e tutti quelli che mi hanno scritto, ai quali spero di essere stato di aiuto.

COMPUTER & RADIO FEST® AMATORE

Organizzazione fiere mostre e mercati

presenta la

FIERA

dell'elettronica & del radioamatore

di Cerea (Verona)

14-15 GENNAIO 2006

ORARIO AL PUBBLICO:
09.00 - 18.00

prevendita biglietti dalle ore 08.00

Informazioni: 337 676719

E-mail: info@compendiofiere.it

MOSTRA MERCATO

- RADIANTISMO
- ELETTRONICA
- COMPUTER
- EDITORIA
- TELEFONIA
- TV-SATELLITARE
- HOBBISTICA
- RADIO D'EPOCA

Comune di Cerea

Seconda parte
n° 245 - Novembre 2005
ISIS - Creazione dei componenti ed editing avanzato
Terza parte
n° 246 - Dicembre 2005
ARES - Tecniche di base per il piazzamento e lo sbroglio di PCB
Quarta parte
n° 247 - Gennaio 2005
La simulazione con VSM

Usare PROTEUS:

Inizia da qui il viaggio alla scoperta di ARES, lo strumento di PROTEUS per lo sbroglio dei circuiti stampati. In questa prima parte vedremo le tecniche di base per il piazzamento e lo sbroglio e alcune tecniche fondamentali per la correzione dei blocchi. Tutte le figure si riferiscono alla versione demo o alla versione completa del programma, se utilizzate la versione lite è possibile che alcune funzioni non siano disponibili. Assumeremo che abbiate installato Proteus nel vostro PC ed eseguito ARES dal menu Start.

UN'ANTEPRIMA DELL'EDITOR

In ARES i menu sono attivati puntando al nome del menu stesso; le varie opzioni sono selezionate cliccando il tasto sinistro del mouse (d'ora in avanti "click sinistro") sull'opzione richiesta. Alcune opzioni possono essere attivate utilizzando i tasti associati ad esse; questi sono visualizzati a destra dell'opzione stessa. Le opzioni che abilitano o disabilitano alcune caratteristiche del programma, mostrano lo stato abilitato di quella caratteristica con un segno di spunta o con una icona in bassorilievo.
L'ampia zona sullo schermo è chiamata Area di *Editing* che agisce come una finestra con veduta parziale dell'intero layout. Potete quindi ritoccare la scala a cui tale veduta è visualizzata usando le opzioni di Zoom nel menu *Visualizza*, oppure con la funzione tasto associata (per default F6 per ingrandire ed F7 per rimpicciolire): in tal caso la posizione del mouse è considerata il nuovo centro dell'area di *Editing*. In alternativa, si può utilizzare la rotellina del mouse per ingrandire o rimpicciolire il disegno attorno l'attuale posizione del puntatore. Potete anche far scorrere il disegno in aree adiacenti tenendo premuto il tasto SHIFT e facendo sbattere il puntatore del mouse contro le pareti dell'area di *Editing*.
La griglia di riferimento sull'area di *Editing* può essere accesa o spenta usando sia il comando Griglia, che tramite la sequenza tasti (per default è "G"). La spaziatura della griglia riflette normalmente il valore di snap, eccetto quando il disegno è ingrandito. In tal caso la spaziatura è un multiplo del valore di snap.
La piccola area in alto a sinistra è chiamata finestra di *Anteprima*. Questa può essere utilizzata anche per vedere l'intero layout. Potete velocemente spostare l'area di lavoro su una sezione desiderata dell'area di *Editing* puntando approssimativamente nella posizione corrispondente, nella finestra di *Anteprima*, eseguendo poi un click sinistro. Il rettangolo verde indica la sezione di area attualmente visualizzata nell'area di *Editing*.
Esattamente sotto l'area di *Editing* c'è il selettore di *Layout* che determina quale piano del disegno è selezionato correntemente oppure consente di selezionarne uno diverso. Il piano attuale si applica sia agli oggetti piazzati che a quelli del PCB.
Si possono scegliere quali piani devono essere visualizzati tramite l'uso del comando *Livelli* del menu *Visualizza*.
Sempre nella parte inferiore dello schermo si possono vedere le coordinate che informano sulla posizione del puntatore in tempo reale,

ARES - Tecniche di base per il piazzamento e lo sbroglio di PCB

di Maurizio Del Corso
m.delcorso@farelettronica.com

quando applicabile. Tali coordinate non si riferiscono all'esatta posizione del puntatore, ma a quella dello snap a cui il puntatore è agganciato. Alla luce di ciò, ne consegue che due sono le cose che hanno effetto sulle coordinate:

- L'attuale dimensione di snap selezionata. L'opzione è disponibile nel menu *Visualizza* ed anche tramite la sequenza di tasti (per default i tasti F1, F2, F3 ed F4). Potete ridefinire i valori di snap tramite il comando *Imposta Griglia* nel menu *Sistema* o ridefinire la configurazione della tastiera con il comando *Imposta mappatura tastiera* del medesimo menu.
- Snap in tempo reale (RTS). Quando tale funzione è abilitata, il cursore si aggancia sui pads o sulle tracce anche quando queste non erano state piazzate con la corrente impostazione di snap. RTS aggancia sempre i pads e i via e può essere disattivata o attivata all'occorrenza usandoli comando *Snap in Tempo reale* nel menu *Strumenti* oppure tramite la sequenza di tasti associata (per default è CTRL+"S").

ARES può essere predisposto per visualizzare una X nella posizione in cui il puntatore è agganciato tramite il comando *X-Cursor*. (la sequenza di default è "X").
Le coordinate possono essere "imperial" o metriche, impostabili tramite il comando *Metrico* (default "M"). Potete anche impostare delle false origini usando il comando Origine (default "O") nel qual caso il colore delle coordinate visualizzate in basso a destra cambia da nero a Magenta.

TECNICHE DI BASE PER IL PIAZZAMENTO E SBROGLIO

Anteprima

Prima di introdurre il concetto di Netlist vogliamo prima fornirvi quei rudimenti necessari per il piazzamento e lo sbroglio di una scheda usando un semplicissimo esempio di scheda campione. La scheda è visualizzata in figura 1.

Scelta del contenitore

Il modo più diretto di costruire un rozzo disegno di una scheda è quello di impostare ARES

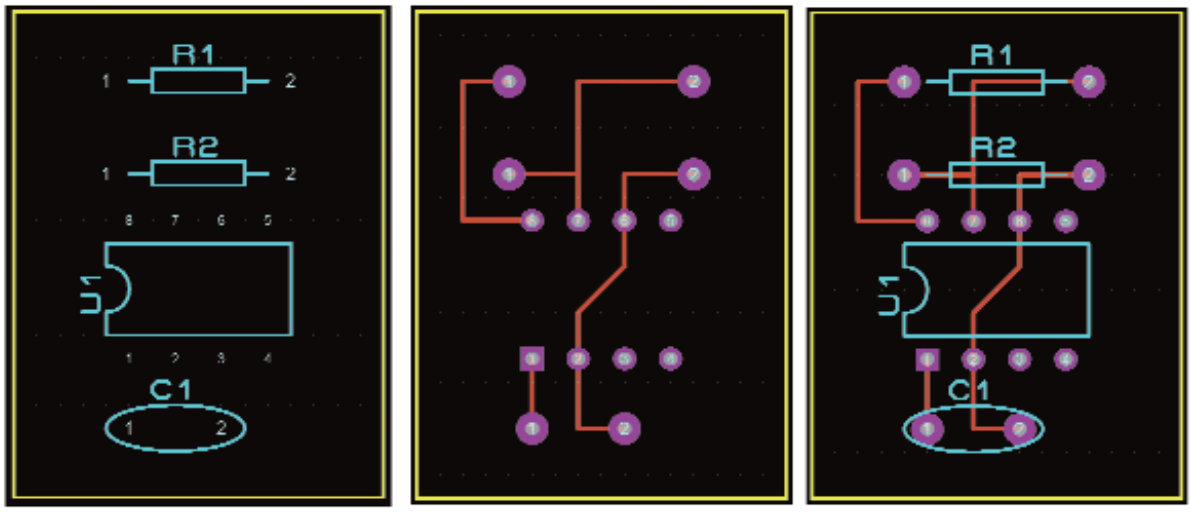


Figura 1 Varie viste del semplice esempio di prova

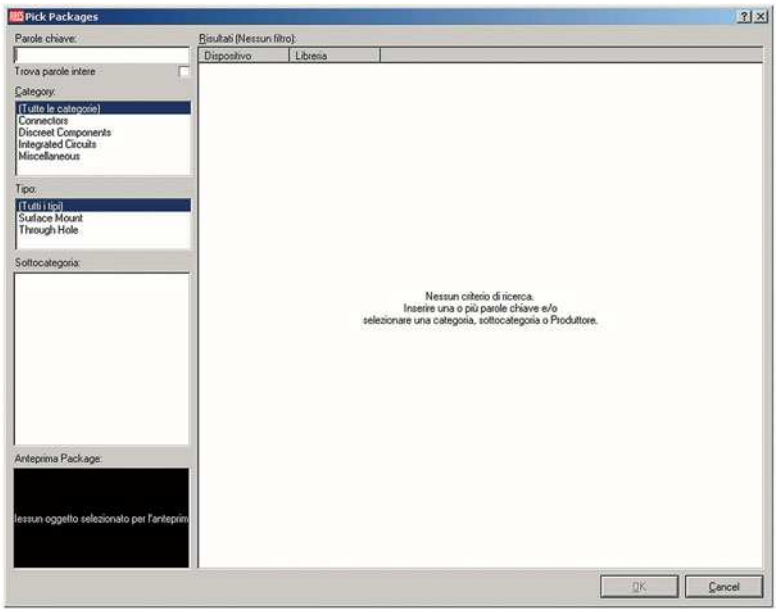


Figura 2 *Finestra Library Pick pronta per la scelta del contenitore*

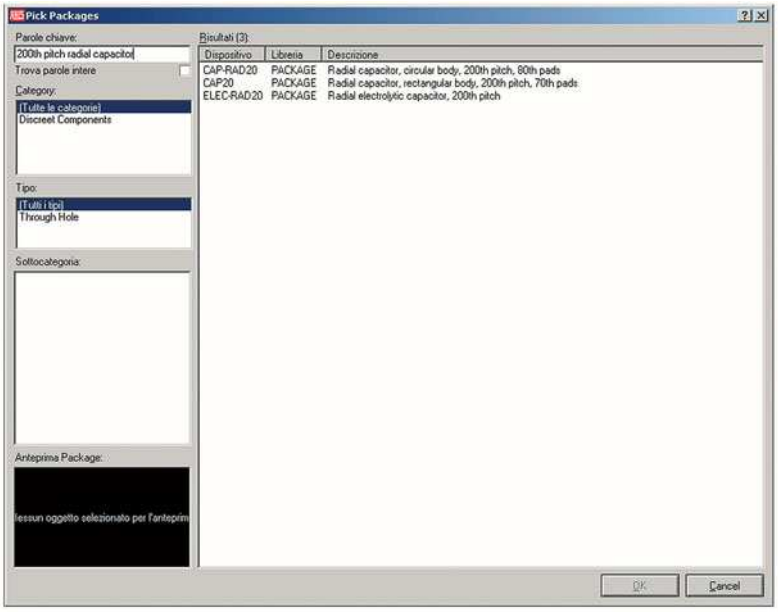


Figura 3 *Finestra Library Pick con il termine di ricerca e CAP20 nei risultati*

nella modalità Package. In questa modalità potete prelevare i simboli dei contenitori dei componenti direttamente dalla libreria e piazzarli nell'area di lavoro.
Nel nostro semplice esempio sono usati tre contenitori:

- CAP20
- DIL08
- RES40

che potete iniziare a prelevare dalla libreria dei contenitori di ARES. Quindi, fate un click sinistro sull'icona Package. In seguito, eseguite un click sinistro sul pulsante "P" in alto a sinistra del *Object Selector*. Apparirà la finestra di dialogo *Library Pick* come mostrato in figura 2.

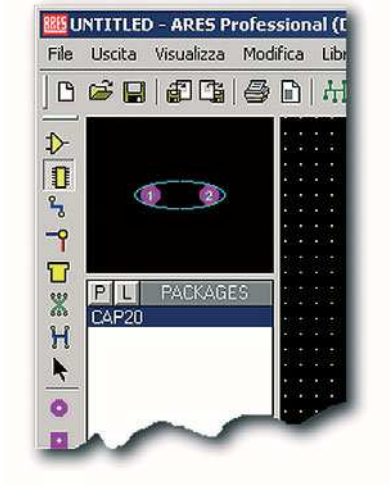


Figura 4 *Object Selector con la parte CAP20 elencata.*

Tale finestra di dialogo è un potente strumento con il quale è possibile cercare una determinata parte in modi differenti.
Probabilmente il modo più semplice (e forse più utile) è semplicemente quello di digitare una parola che descrive la parte che stavate cercando. Ciò, in modo del tutto simile al motore di ricerca Google™ su Internet, ricerca una parte nella libreria sulla base della descrizione da voi fornita. Provate ora con il condensatore, digitando il termine di ricerca "200th pitch radial capacitor" nel campo *Parole chiave*. Dovreste vedere che la parte CAP20 è elencata nei risultati della ricerca come mostrato nella figura 3.

Fate doppio click sulla parte CAP20 per prelevarla nel *Object Selector* così come mostrato nella figura 4.

Potreste naturalmente ripetere il suddetto procedimento per i rimanenti contenitori, tuttavia è istruttivo usare qui gli altri metodi differenti

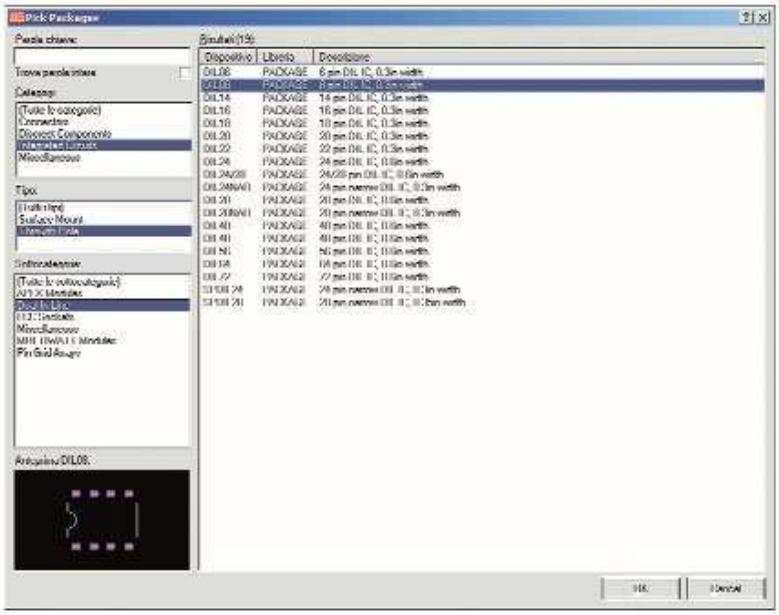


Figura 5 Dialogo Library Pick con i filtri impostati e i relativi risultati

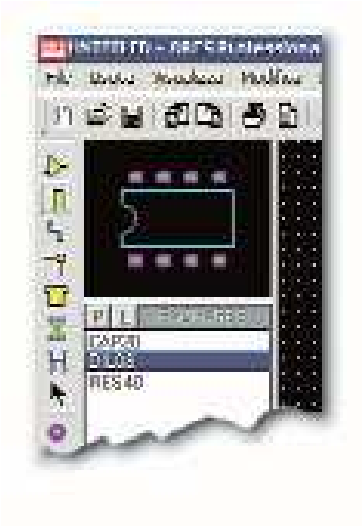


Figura 6 Object Selector con i contenitori richiesti selezionati

di ricerca. Al di sotto della casella di testo *Parole chiave*, sul lato sinistro della finestra di dialogo trovate una quantità di filtri di ricerca che possono essere impiegati indipendentemente per raffinare la ricerca. Tali filtri sono particolarmente utili per elencare le librerie quando non siete del tutto sicuri del termine di ricerca più adatto per la parte desiderata. Provate tele eventualità con la parte DIL08, cancellando tutto il testo dalla casella *Parole chiave* e poi selezionando la categoria *Integrated Circuits*, il tipo *Through Hole* e la sotto categoria *Dual Inline* (figura 5).
Facendo doppio click sul risultato DIL08 preleverà, come di consuetudine, la parte per essere usata nel disegno.

Infine, useremo una combinazione dei suddetti procedimenti per trovare e prelevare la parte RES40. Scegliete *Discrete Components* come categoria e digitate RES40 nella casella *Parole chiave*. In tale situazione il termine è ricercato solo in quelle parti che attengono al sottoinsieme filtrato dal campo categoria e tipo. Selezionate la parte nel modo consueto e chiudete la finestra di dialogo.

Piazzamento delle parti

I tre nomi dei contenitori dovrebbero ora apparire nel *Object Selector* come mostrato in figura 6. L'ultima parte prelevata viene evidenziata.

Inizieremo a piazzare la parte DIL08 sul disegno. Assicuratevi che questa parte sia selezionata eseguendo un click sinistro su essa nel *Object Selector*. Sarà mostrata un'anteprima della parte nella finestra di *Anteprima* con l'orientamento attuale. Prima del piazzamento, potete usare le icone rotation (mostrate in figura 7) per ruotare la parte nel desiderato angolo di orientamento.

Figura 7 Le icone rotation di ARES

Per piazzare la parte puntate in qualche punto centrale del disegno nell'area di *Editing* e premete (e mantenete premuto) il pulsante sinistro del mouse: in quel punto apparirà il contorno di un IC a 8 pin, colorato di verde. Potete trascinarlo con il mouse e posizionarlo approssimativamente al centro dello schermo e quindi rilasciare il pulsante.
Ora selezionate il contenitore RES40 e piazzate due di queste parti, a distanza di 0.1" (2 quadratini di griglia) esattamente sopra l'integrato ad 8 pin. In modo simile, piazzate il condensatore esattamente sotto il pin 1.
A meno che non siate già degli esperti, le parti

potrebbero non essere correttamente posizionate, il che ci consente di dare uno sguardo ai procedimenti per spostarle. Puntate sul contorno di un componente (o ai suoi pin), e fate un click destro per evidenziale la parte. Ora, sempre tenendo il puntatore sulla parte e tenendo premuto il pulsante sinistro, trascinate il puntatore. Questo è uno dei modi per spostare un oggetto. Potete anche cancellare un oggetto evidenziato cliccandovi sopra con il pulsante destro. Tramite il menu *Modifica* (o come tasto pre-configurato CTRL+Z) è possibile sempre annullare l'operazione.
Tutti gli oggetti evidenziati possono essere posti nella modalità normale puntando in una zona libera da oggetti quindi effettuando un click destro.

Sbroglio

La modalità di sbroglio si inizia cliccando con il pulsante sinistro sull'icona Piste. *Object Selector* cambierà aspetto per visualizzare un elenco di stili di Traccia o piste (la scelta di default è la larghezza T20 per una pista di 20 millesimi di pollice).
Le Piste sono piazzate con un click sinistro per ciascun punto lungo l'intero percorso desiderato, cliccando il pulsante destro quando terminato. Ora provate a sbrogliate la scheda manualmente, così come mostrato precedentemente in figura 1. Assicuratevi di iniziare e terminare una pista su una piazzola. Il cursore del mouse si trasforma in una piccola "x" quando il mouse è direttamente sopra una piazzola ad indicare che è possibile iniziare o terminare una pista (figura 8).

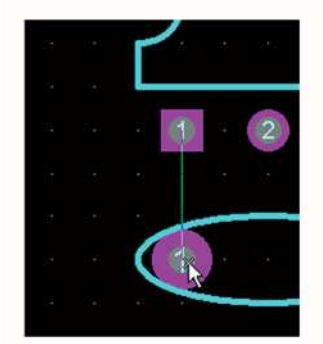


Figura 8 *Il cursore del mouse sopra una piazzola prima della terminazione di una pista*

Figura 9 *La finestra Modifica Componente con un esempio di Part ID e Valore digitati per un resistore*

Altri punti di cui tenere conto durante la modalità di sbroglio sono:

- Fare un doppio click in uno stesso punto, determina il piazzamento di una via e cambia il piano di layout come definito nel comando *Imposta coppie di livelli* nel menu *Sistema*. I vari tipi di via possono essere visualizzati cliccando sull'icona Via.
- Durante lo sbroglio, è possibile cambiare il piano di layer premendo i tasti PGUP e PGDN. Inoltre CTRL-PGUP seleziona il piano superiore mentre CTRL-PGDN sceglie quello inferiore.
- Tenendo il tasto CTRL premuto mentre si sbroglia manualmente una pista permette di piazzare segmenti di piste curve. Il verso e l'ampiezza di curvatura dell'arco (orizzontale\verticale o viceversa) è determinato nel modo più naturale immaginabile da come allontanate il mouse da dei punti fissi. Nella progressione delle operazioni è meglio tenere premuto il tasto CTRL, quindi spostare il mouse, poi fare il click sinistro ed infine rilasciare il tasto CTRL.

Annotazione

Quando sono piazzati i componenti nella modalità Package, essi non hanno alcuna informazione associata (più avanti vedremo come i componenti sono annotati automaticamente usando la Netlist).
Per annotare i componenti, scegliete l'icona "Modifica componenti" quindi eseguite un click sinistro su ciascun componente a rotazione. Per ciascun componente cliccato appare una finestra che dispone di campi sia per l'ID della parte che per il suo valore (figura 9). Potreste trovare più semplice l'uso della tastie-

ra durante l'annotazione: i tasti cursore sposteranno il puntatore di una unità di griglia e il tasto ENTER si comporterà come il click sinistro del mouse o come il pulsante OK nella finestra di dialogo. Inoltre il tasto TAB farà spostare di un campo quando editiamo nella finestra di dialogo.

In alternativa, può essere usato il comando *Generazione automatica Nomi* per generare delle sequenze numeriche per annotare componenti.
Sia l'etichetta ID che il valore possono essere spostati evidenziando il componente che le contiene quindi puntando specificatamente su ID o valore ed infine trascinandole usando il pulsante sinistro del mouse. Tenete ben presente che possono essere selezionate differenti griglie di snap sia tramite il menu *Visualizza* che con la sequenza di tasti associata (per default queste sono F1, F2, F3 e F4 ma le associazioni dei tasti ai comandi possono essere configurate tramite il comando *Imposta mappatura tastiera* nel menu *Sistema*).
La dimensione di default di queste etichette può essere determinata usando il comando Imposta Modello nel menu Sistema.

Bordo della Scheda

ARES possiede uno speciale piano, il piano EDGE, che serve ad identificare il bordo della scheda. Tale oggetto, costituito da elementi grafici 2D, è piazzato sul piano EDGE ed apparirà nei piani di costruzione della scheda su ciascun altro piano.
Nel caso specifico, il bordo della scheda è un semplice rettangolo.

Per piazzare un bordo rettangolare:

1. Premete l'icona Box.
2. Selezionate EDGE dal selettore di piano *Selezione Livelli*.
3. Puntate laddove volete l'angolo superiore sinistro del vostro bordo, premete il pulsan-

Scheda easyPIC3

La rivoluzionaria scheda di sviluppo per PICmicro

- ✔ Programmatore USB2.0 on-board
- ✔ Tastiera a 32 tasti
- ✔ 32 LED per il monitoraggio degli I/O
- ✔ 4 cifre LED a 7 segmenti
- ✔ Predisposizione per moduli LCD alfanumerici
- ✔ Predisposizione per moduli LCD grafici
- ✔ Predisposizione per comunicazione RS232
- ✔ Predisposizione per tastiera PS2
- ✔ Predisposizione per sensore di temperatura DS1820
- ✔ Supporto per tutte le famiglie PIC (anche PIC10F)
- ✔ Predisposizione per comunicazione USB
- ✔ Alimentazione esterna o via USB
- ✔ Fornita con 16F877
- ✔ Disponibile con o senza display

Ordinala subito su **www.farelettronica.com** oppure telefona allo 02.66504794

te sinistro del mouse e trascinatelo fino a racchiudere le parti nel disegno.

4. Se avete necessità di ridimensionare il bordo, evidenziatelo puntandovi il mouse e quindi fate un click destro ed infine, usando il pulsante sinistro del mouse, trascinate le maniglie di dimensionamento per la grandezza desiderata.

Sono pienamente supportati i bordi di forma irregolare e curva; questi sono formati da serie di linee ed archi oppure selezionando l'icona *Percorso grafico* e disegnando una forma chiusa.
Quando avete concluso di sperimentare i comandi con il presente esercizio, salvatelo se volete e iniziate il prossimo esercizio usando il comando *Nuovo Layout* nel menu file.

TECNICHE DI CORREZIONE DEI BLOCCHI

Abbiamo già visto che un oggetto può essere evidenziato puntandovi sopra il mouse e facendo un click destro e che, una volta evidenziato, può essere trascinato (pulsante sinistro) o cancellato (pulsante destro).
C'è però anche un altro modo per evidenziare oggetti. Se, puntando in una zona libera da oggetti, tenete premuto il pulsante destro ed in queste condizioni trascinate il mouse, dovreste vedere un rettangolo verde apparire sullo schermo.Quando rilasciate il pulsante, ciascun oggetto racchiuso nel rettangolo sarà evidenziato. Caricate il file "PPSU.LYT" dalla directory *Samples* della vostra installazione di Proteus (posta nella sotto-cartella *Tutorials*) e provate a trascinare un rettangolo di selezione attorno tutta la scheda. Quando avete evidenziato gli oggetti entrano in gioco le icone *Copia*, *Sposta*, *Ruota* ed *Elimina*.
Provatene gli effetti, tenendo conto che:

- Il tipo di oggetti che può essere evidenziato può essere cambiato attraverso il comando *Filtro Selezione* (per default associato con CTRL+X). Ciò permette anche di scegliere su quale piano gi oggetti saranno evidenziati.
- Potete ripristinare un'operazione eseguendo il comando *Annulla* (default associato con "CTRL +Z").
- Quando eseguite la rotazione di un intero blocco, vi sarà richiesto di fissare un origine attorno al quale avverrà la rotazione\riflessione.

Provate a completare lo sbroglio di qualche pista ed evidenziando uno o due componenti, provate a spostarli.
Ricordate che per editare un componente dovete prima selezionare l'icona *Package* oppure l'icona *Modifica componenti*, per spostare un componente dovete prima selezionare l'icona *Package* e per piazzare le piste dove selezionare prima l'icona *Piste*.

TECNICHE DI CORREZIONE DELLO SBROGLIO

Una delle maggiori caratteristiche di ARES è la potente capacità di editare lo sbroglio.
Diversamente da molti programmi per il disegno di PCB, l'editing dello sbroglio in ARES si basa sulla topologia attuale piuttosto che dipendere da come sono state configurate le sezioni delle piste. Inoltre, le modifiche possono essere eseguite in ogni parte dello sbroglio e non solo sulle sezioni tra nodi.

Piazzamento dei Via

In quasi tutte le circostanze i via (piazzole di comunicazione tra diversi livelli) vengono piazzati automaticamente. Selezionate l'icona *Piste* e piazzate un segmento di pista con un click sinistro su due punti.
Ora eseguite un click sinistro sopra il secondo punto per la seconda volta e quindi sul terzo ed infine un click destro per terminare la pista. Cliccando (sinistro) due volte su uno stesso punto fa sì che ARES piazzi un via e cambi il corrente piano in modo tale che il prossimo segmento di pista sia sbrogliato sul piano rame. Contemporaneamente un via viene piazzato nel punto in cui il percorso commuta il piano.
Il comando *Imposta coppie di livelli* nel menu *Sistema* determina quale piano viene cambiato. È possibile definire coppie, triplette di piani, o qualsiasi altra combinazione necessaria. Potete anche cambiare il piano manualmente mentre sbrogliate la pista usando i tasti PGDN e PGUP, ma in questo caso il via non viene piazzato.

Il tipo di via utilizzato può essere scelto dal *Object Selector* selezionando l'icona *Via* e scegliendo uno dei vari stili di via disponibili. In questa modalità potete anche piazzare, sostituire, evidenziare, spostare e cancellare vias. Il pulsante sinistro e destro del mouse opererà esattamente come per l'editing dei componenti.
Per schede a piani multipli, potete selezionare via comunicanti, ciechi o interni (sepolti) tramite l'uso dell'icona *Via* e quindi cambiando il piano con *Selezione Layer*.

Cambiare la larghezza di pista

Se avete bisogno di cambiare la larghezza di una sezione della pista potete seguire due diversi approcci:

- Selezionate semplicemente lo stile di pista richiesto e piazzare la nuova pista sulla vecchia.
- Evidenziare la pista con il pulsante destro del mouse quindi cliccando una seconda volta ed infine selezionando dal menu contestuale il richiesto stile di pista.

Rammentate che per cambiare la larghezza di pista in modo globale, potete sempre editare lo stile di pista appropriato selezionandolo e quindi premendo il pulsante "E" nel *Object Selector* (figura 10). Notare che questo cambia la larghezza di tutte le piste aventi lo stile di pista editato.

Restringimento automatico della pista (Auto Necking)

Il motivo principale nel restringere una pista è quello di farla passare tra due piazzole od altri ostacoli senza violare le regole di progetto. La modalità di auto restringimento di ARES esegue automaticamente questa operazione.
La funzione è controllata dal comando *Imposta Regole predefinite* nel menu *Sistema*.
La finestra di dialogo vi permette di digitare lo spazio tra piazzole, piazzola-pista, tra le piste ed inoltre lo stile di restringimento automatico. Il valore di default è T10 (10 millesimi di pista).

Evidenziare una pista

Per riposizionare, cancellare o copiare una

sezione di pista vi è necessario evidenziarla (cosa comune a qualsiasi manipolazione di oggetti in ARES).
Generalmente l'evidenziazione avviene cliccando con il pulsante destro sull'oggetto, ma in questo caso vi sono alcune sottigliezze che vale la pena analizzare:

- ARES evidenzierà le piste solo sull'attuale piano selezionato con il comando *Selezione Livelli*, quindi selezionate il piano desiderato come attuale.
  La barra spazio o il pulsante centrale del mouse (se ne è dotato) causerà la selezione del prossimo piano nella sequenza definita con il comando *Imposta coppie di livelli*. In alternativa, potete spostarvi su e giù con i tasti PGUP and PGDN.
- Se cliccate su un via, oppure su un punto di giunzione di diverse piste, tutte le piste che fanno capo a quel punto vengono evidenziate.
- Se cliccate con il pulsante destro del mouse su una pista e la trascinate mentre tenete il pulsante premuto, potete selezionare una sotto-sezione di quella pista.

Come per qualsiasi altro oggetto da editare, puntando in una zona vuota e premendo quindi il pulsante destro, tutte le piste evidenziate ritorneranno nella modalità normale.
Vi suggeriamo di acquisire confidenza nel selezionare ed evidenziare le piste di uno sbroglio prima di proseguire - è importante, anche se per niente difficoltoso, avere una buona pratica sul funzionamento di tali caratteristiche.
Con una sezione di pista evidenziata, potete trascinarne qualsiasi segmento puntando ad un punto e trascinandolo con il pulsante sinistro del mouse.

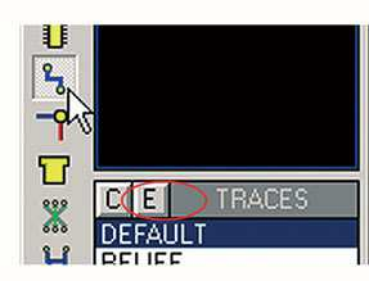


Figura 10 L'icona modalità Piste e il pulsante Modifica Stile (E)

Figura 11 Menu di Contesto per piste evidenziate

Cancellazione di una pista evidenziata

Ci sono due modi per cancellare una pista:

- L'icona *Elimina Blocco* cancellerà tutti gli oggetti evidenziati sul disegno. Durante l'editing dello sbroglio tali oggetti sono tipicamente le piste che avete evidenziato, quindi questo è un modo rapido per cancellare piste.
- Potete anche evidenziare una sezione singola di pista interconnessa usando il menu di contesto.

Menu di contesto

Se su una pista evidenziata eseguite un click destro, apparirà un menu a scomparsa (figura 11) con le seguenti opzioni:

- Elimina, Copia e Sposta.
- Modifica il piano (Livello Pista) e stile di pista (Stile pista).
- Modifica lo stile del via (Stile via)

Il comando *Copia* fornisce un modo per costruire piste appaiate come per i bus di memoria.
Potete fare quante copie volete cliccando il pulsante sinistro ripetutamente. Fate un Click destro per terminare.

Riposizionamento dello sbroglio

Infine, c'è anche un altro modo per modificare il percorso di una pista.
Evidenziata la pista, potete alterarne il percorso posizionando semplicemente una nuova sezione di pista (nel modo usuale) che inizia e finisce su quella vecchia.
ARES calcola automaticamente quale sezione è cortocircuitata e la rimuove. Notare anche che i vias che si rendono ridondanti (vias secondari) saranno rimossi, insieme con le sezioni cortocircuitate.
Durante il processo di sbroglio, troverete tali funzioni estremamente utili per spostare piste che ne bloccano altre.

Verifica della Connettività

Anche se non strettamente connessa con le opzioni di editing dello sbroglio, la verifica della connettività risulta molto utile verso la fine del processo di sbroglio per controllare quali sono le piste connesse e (più importante) quelle non collegate (ad esempio, potreste voler sapere se un segnale di clock è stato connesso a tutti gli integrati che lo richiedono).
Per vedere in opera tale caratteristica, selezionate l'icona *Evidenziazione Connettività*.
Cliccando con il pulsante sinistro del mouse su una piazzola di un componente con una pista connessa, vedrete che tutto ciò che è connesso con quella piazzola si illumina.
Tutti questi oggetti evidenziati rimarranno illuminati anche durante uno zoom o lo scorrimento, fino a che non eseguite il comando *Ridisegna* (default "R").
Nella modalità di verifica della connettività, cliccando su una net presente nell'elenco e premendo il pulsante "T" del selettore, la net corrispondente e tutti gli oggetti ad essa collegata si illumineranno.
Cliccando l'icona *Elimina* ciò cancellerà le piste o i vias illuminati e ciò consentirà di cancellare tutte o parte delle nets.

CONCLUSIONI

A questo punto avete le nozioni necessarie per poter iniziare a lavorare con ARES.
Dedicheremo la prossima puntata allo sbroglio (automatico e manuale) di PCB mediante l'uso di una netlist, quindi la possibilità di interagire con ISIS oltre a mostrare come creare componenti personalizzati da inserire nelle proprie librerie.
Vi ricordiamo che per tutte le vostre perplessità, idee e commenti potete usare il forum dedicato a PROTEUS sul sito www.farelettronica.com

CCTV NEW ENTRY

QUAD PROCESSOR DIGITALE A COLORI

Completo quad processor real-time a colori in grado di suddividere lo schermo di un monitor in quattro zone, visualizzando le immagini provenienti da 4 telecamere. Visualizza a schermo intero un ingresso specifico ed effettua la scansione degli ingressi programmati a velocità regolabile. Picture in picture. Adattatore 12V/600mA (incluso); dimensioni: 230x195x48mm.

VQSM4CRT **€ 205,00**

QUAD COMPRESSOR B/N

Modulo quad B/N, suddivide lo schermo di un monitor in quattro parti, visualizzando le immagini provenienti da 4 telecamere in real time. Risoluzione: 720 x 576 pixel; rinfresco dell'immagine: 25/30 campi al sec.; On Screen Display; alimentazione 12Vdc - 6W; dimensioni: 240 x 150 x 45mm. Interfacciabile con impianti di registrazione. Alimentatore non compreso.

FR118 **€ 85,00**

COMMUTATORE VIDEO 8 CANALI

Possibilità di funzionamento manuale o automatico con selezione dei canali attivi. In modalità automatica è possibile scegliere la velocità di commutazione. Ingressi video: 8 (connettore BNC); uscita video: 1 (connettore BNC); sensibilità ingressi video: 1Vp-p / 75 ohm; alimentazione: 12V DC - 400 mA (adattatore non compreso); dimensioni: 265 x 190 x 55mm.

VMS8 **€ 32,00**

REGISTRATORE A/V WIRELESS

Sistema multimediale senza fili operante sulla banda dei 2,4 GHz composto da un registratore audio/video con display LCD a colori da 2,5 pollici e da una telecamera CMOS a colori con audio nascosta all'interno di una vera penna. Il dispositivo è dotato di interfaccia USB tramite cui è possibile eseguire il download delle registrazioni da PC. Può essere utilizzato anche per visualizzare immagini in formato JPG, per riprodurre filmati di tipo ASF e come lettore MP3. Viene fornito completo di CD-Rom che include il programma per la gestione delle funzioni multimediali. Alimentazione: mediante batteria ricaricabile al Litio (inclusa), adattatore di alimentazione 220 Vac/5 Vdc 1 A (incluso) o mediante adattatore per batterie di tipo AA (non incluse).

FR290 **€ 660,00**

DVR 4 CANALI CON HARD DISK 120 GB E BACK-UP CON COMPACT FLASH

Innovativo registratore digitale video (DVR) a quattro canali completo di Hard Disk da 120 GB con cassetto estraibile e con possibilità di effettuare back-up su Compact Flash.

Formato Video: NTSC/PAL; compressione: MPEG4; ingressi video: 4 canali (connettori BNC); uscite video: 2 (Video OUT, VCR OUT), quattro modalità di registrazione; modalità di riproduzione: standard avanti e indietro, veloce avanti e indietro, frame, zoom in; funzioni di ricerca: telecamera, data&ora; alimentazione: 12VDC/4A (adattatore incluso); potenza assorbita: 20W; dimensioni: 430 x 305 x 77mm. È disponibile separatamente un cassetto estraibile supplementare senza Hard Disk (cod. DVRCARTR2).

DVR4QAF-120 (DVR con HDD) **€ 628,00**
DVRCARTR2 (cassetto supplementare) **€ 52,00**

TELECAMERA CCD A COLORI DA ESTERNO

Telecamera CCD a colori resistente agli agenti atmosferici munita di custodia in alluminio e staffa per il fissaggio. Viene fornita completa di adattatore da rete. Elemento sensibile: 1/4" CCD a colori; risoluzione orizzontale: 420 linee TV; sensibilità: 0,8 lux (F1.2); ottica: f3.6 mm; alimentazione: 12 Vdc / 400mA (alimentatore stabilizzato incluso); dimensioni: Ø34 x 77 mm.

CAMCOLBUL4L **€ 110,00**

TELECAMERA CCD B/N DA ESTERNO

Telecamera CCD bianco/nero resistente agli agenti atmosferici munita di custodia in alluminio e staffa di fissaggio. Viene fornita completa di adattatore da rete. Elemento sensibile: 1/3" LG B/W CCD; risoluzione orizzontale: 420 linee TV; sensibilità: 0,05 lux (F1.2); ottica: f3.6 mm; alimentazione: 12 Vdc / 400mA (alimentatore stabilizzato incluso); dimensioni: Ø34 x 77 mm.

CAMZWBUL4L **€ 73,00**

TELECAMERA PER VISIONE POSTERIORE PER AUTOVEICOLI CON MIRROR

Telecamera CMOS a colori per visione posteriore adatta per essere installata su qualsiasi autoveicolo. Consente di avere sempre un'ottima visuale sia in fase di retromarcia che durante manovre difficoltose effettuate in spazi particolarmente limitati. Sensore: 1/3" CMOS a colori; risoluzione: 380 linee TV; sensibilità: 1,5 lux / F2; ottica: f 6mm; apertura angolare: 52°; alimentazione: 12 Vdc / 100mA max. (stabilizzata); adattatore di rete incluso; dimensioni: 56 x Ø30-24mm.

CAMCOLBUL6C **€ 52,00**

VIDEOCITOFONO B/N COMPLETO

Sistema videocitofonico bianco/nero comprendente una unità esterna con microfono parla/ascolta, pulsante di chiamata e un'unità interna completa di cornetta. E' possibile espandere il sistema con una unità interna supplementare (CAMSET14MON).
Unità interna: Monitor: 4" bianco/nero CRT tipo flat; risoluzione: migliore di 380 linee TV; consumo: 13W/25W in uso, 4W/7W in standby; alimentazione: 230VAC.
Unità esterna: Telecamera: sensore 1/3" CMOS; ottica: 3.6mm con apertura angolare di 78°; sensibilità: 0,1Lux; illuminatore IR (portata circa 2 metri).

CAMSET14 **€ 120,00**
CAMSET14MON (unità supplementare) **€ 78,00**

MONITOR TFT 8" 16:9

Monitor con display TFT LCD da 8 pollici a colori con altoparlante incorporato. Dispone di 2 ingressi video analogici e di un ingresso audio. Sistema di funzionamento: PAL/NTSC con selezione automatica. Regolazioni immagine; telecomando; 2 ingressi video: AV1/AV2; 1 ingresso audio: AV1; retroilluminazione: CCFT; luminosità: 350 nits; risoluzione: 1140(H) x 234(V); alimentatore 11-14 Vdc non incluso; consumo: 800mA/10W; dimensioni: 200 x 135 x 33mm. Viene fornito completo di supporto da tavolo e di telecomando a infrarossi.

MONCOLHA8 **€ 215,00**

FUTURA ELETTRONICA

Via Adige, 11 - 21013 Gallarate (VA)
Tel. 0331/799775 - *www.futuranet.it*

Maggiori informazioni su questi prodotti e su tutte le altre apparecchiature distribuite sono disponibili sul sito *www.futuranet.it* tramite il quale è anche possibile effettuare acquisti on-line.

CONTENITORE A TENUTA STAGNA

Contenitore metallico con vetro frontale, mascherina anti riflesso, completamente stagno e riscaldato tramite alimentazione da rete a 220 volt. Permette di alloggiare comodamente le telecamere da sorveglianza mod. FR110 e FR111 o simili; possibilità di fissaggio a muro tramite la staffa con snodo non inclusa nella confezione.

FR112 **€ 32,00**

STAFFA PER CONTENITORI

Staffa metallica con snodo adatta ad essere utilizzata col contenitore stagno FR112. Carico massimo 10 Kg, lunghezza 205 mm, angolo di rotazione 90 gradi, peso 800g.

FR113 **€ 11,00**

FALSA TELECAMERA IN METALLO

Perfettamente uguale in ogni particolare ad una telecamera vera! Il contenitore metallico a tenuta stagna consente di utilizzare la falsa telecamera all'esterno o all'interno. Contenitore: metallo verniciato. Alimentazione Led: Batteria 1,5V (batteria non compresa); dimensioni: 250 x 120 x 60 mm (incluso braccio); fissaggio a muro: 4 tasselli (compresi).

FR223 **€ 24,00**

FALSA TELECAMERA PLASTICA DA INTERNO

Corpo ed obiettivo in plastica, alimentazione mediante 3 pile a stilo. La falsa telecamera dispone di un sensore di movimento che la attiva quando qualcuno passa davanti all'obiettivo. Durante il periodo di attivazione (che dura circa 20 secondi) il corpo ruota ed il led lampeggia. Alimentazione: 3 x 1,5V AA (batterie non comprese); altezza: 170mm circa.

FR223P **€ 6,00**

FALSA TELECAMERA MOTORIZZATA

Falsa telecamera per applicazioni da interno/esterno dotata di sistema di rotazione motorizzato. Completa di led lampeggiante. Corpo in metallo che conferisce al sistema un aspetto del tutto simile ad una vera telecamera. Viene fornita con alimentatore da rete e 20 metri di cavo. Possibilità di regolare l'angolo di rotazione tra 22,5 e 350 gradi. La telecamera ruota per 30 secondi ogni tre minuti.

FR234 **€ 56,00**

FALSA TELECAMERA DOME

Falsa ma realistica telecamera dome da interno. Dimensioni: Ø87 x 57mm, peso: 66g.

CAMZWDH1 **€ 10,00**

Tutti i prezzi s'intendono IVA inclusa.

Ventiduesima parte
n° 244 - Ottobre 2005
Tecniche di debug

Ventitreesima parte
n° 245 - Novembre 2005
Gestione delle interruzioni

Ventiquattresima parte
n° 246 - Dicembre 2005
Sistemi operativi

Vitamina C:

In questa ultima puntata di Vitamina C prenderemo in considerazione i sistemi operativi. Vedremo cosa sono, come funzionano, e come scrivere i nostri programmi per utilizzarli al meglio.

INTRODUZIONE

Fino ad ora abbiamo considerato programmi singoli ed autosufficienti, cioè programmi che venivano eseguiti da un processore o microcontrollore, e che avevano il completo controllo del flusso di esecuzione e dell'hardware del sistema. Moltissimi software per piccoli sistemi embedded sono scritti in questo modo. Come abbiamo visto nella scorsa puntata tutti i compiti sono gestiti da un loop principale, ed in parte dalle routine di interruzione. Se però i compiti da svolgere sono molti, le loro interazioni complesse, e le temporizzazioni abbastanza critiche, diventa molto difficile gestirli nel modo già visto. Si pensi ad esempio ad un sistema di controllo industriale che deve gestire, oltre ad un complesso e delicato processo, anche una comunicazione di rete basata sul protocollo TCP/IP, un'interfaccia utente grafica e le relative periferiche di input ed output. In questi casi l'uso di un sistema operativo (*Operating System*, in breve OS) potrebbe semplificare molto lo sviluppo del software, renderlo anche più modulare, efficiente e robusto.
Al contrario di quello che si potrebbe pensare, un sistema operativo non è necessariamente uno strumento grosso e complesso (come quelli utilizzati sui personal computer). Esistono dei sistemi operativi piccolissimi e molto semplici che possono essere eseguiti perfino su microcontrollori ad 8 bit! Una volta compresi i principi su cui si basa il loro funzionamento non è neanche eccessivamente difficile scriverne uno secondo le proprie esigenze. Un OS svolge diverse funzioni: in primo luogo permette di eseguire "contemporaneamente" più programmi, o di svolgere più compiti, in secondo luogo si occupa della gestione di basso livello dell'hardware, mettendo a disposizione dei programmi un'interfaccia più astratta e fornendo anche diversi strumenti utili per facilitare la scrittura. Le caratteristiche e le funzioni messe a disposizione dipendono dal particolare sistema operativo, e possono variare moltissimo da un prodotto all'altro. Ovviamente qui verranno prese in considerazione quelle di base, comuni a quasi tutti i sistemi operativi.

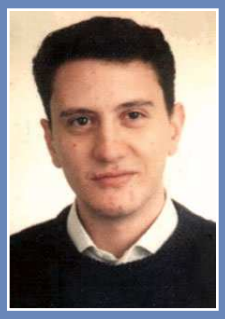
COME FUNZIONA UN SISTEMA OPERATIVO

Come già detto esistono sistemi operativi complessi che offrono funzioni avanzate, come ad esempio il supporto per le periferiche di archiviazione di massa, un file system, protocolli di comunicazione (stack TCP/IP), ed il supporto per le periferiche grafiche. Ovviamente sistemi operativi di questo tipo risultano molto più costosi ed esigenti in termini di risorse, ed in certi casi più complessi da utilizzare. Tutti i sistemi operativi comunque sono basati sugli stessi concetti e meccanismi di base, che verranno descritti di seguito.

Task

Uno dei vantaggi principali offerti da un sistema operativo è la possibilità di eseguire simultaneamente routine diverse ed indipendenti. Ciascuno di queste routine viene definita **task** (che in inglese significa "*compito*"). Ovviamente l'esecuzione dei task non è realmente simultanea, ma essi vengono eseguiti a turno condividendo così le risorse dell'unica CPU. É il sistema operativo che decide momento per momento quale task deve essere eseguito, e quale restare in attesa. I task possono alternarsi molto velocemente, e quindi il risultato è simile a quello che si avrebbe se venissero eseguiti davvero in concorrenza. I vari task sono program-

Sistemi operativi



di Antonio Di Stefano
a.distefano@farelettronica.com

mi del tutto indipendenti tra loro, ed a ciascuno sono associati contesti e parametri diversi. Ad esempio ciascun task ha un suo stack, un suo identificativo per distinguerlo dagli altri, ed una sua priorità di esecuzione assegnata dall'utente.
Quando un task viene momentaneamente sospeso in favore di un altro, vengono conservate tutte le informazioni necessarie per potere riprendere l'esecuzioni nelle stesse condizioni in cui la si è lasciata. Questo meccanismo è simile al salvataggio del contesto che avviene durante le interruzioni: verrà salvato ad esempio il Program Counter, lo status register, i vari registri, e verrà cambiato lo stack.
Ogni task può trovarsi in diversi stati di esecuzione: può essere correntemente in esecuzione (*running*), bloccato in attesa di qualche evento (*waiting*), o pronto per l'esecuzione, ma non eseguito in quel momento (*ready*). La figura 1 rappresenta questi stati e le loro relazioni. Un task si trova nello stato di attesa (waiting) se sta attendendo che si verifichi un evento esterno, o se ha l'esigenza di restare in "pausa" per un certo periodo di tempo (perché non ha niente da fare). Fino a quando un task si trova nello stato di attesa il sistema operativo non lo considererà ai fini dell'esecuzione, però controllerà se si sono verificate le condizioni per riportarlo nello stato ready (si è verificato l'evento esterno atteso, o è trascorso il tempo di attesa specificato). I task nello stato ready sono quelli che di fatto si contendono l'esecuzione, e quindi lo stato running (in cui ovviamente si può trovare un solo task alla volta). La scelta del task da eseguire, tra quelli ready dipende dagli algoritmi di scheduling utilizzati dall'OS.

Lo scheduler

Il cuore di ogni sistema operativo è lo scheduler, è lui infatti a gestire la scelta dei task da eseguire, a cambiarne lo stato, o a stabilire se un task deve essere sospeso in favore di un altro a priorità maggiore. L'algoritmo impiegato dello scheduler per stabilire quale task eseguire in un dato momento, determina gran parte delle caratteristiche dell'intero sistema operativo. Esistono diversi possibili algoritmi di scheduling (come il "*first in first out*" o lo "*shortest job first*", giusto per citarne due famosi), ma quello che più si avvicina alla descrizione che è stata fatta fino ad ora, e che offre più possibilità, è il "**round robin**" e le sue varianti. In questo caso i task nello stato ready sono eseguiti circolarmente, uno dopo l'altro. Se si utilizza un approccio **cooperativo** (detto anche "**non-preemptive**", o senza prelazione) allora sarà il task in esecuzione (e solo lui) a decidere di sospendersi quando avrà completato il suo compito, passando nello stato waiting o ready e dando così l'opportunità agli altri di essere eseguiti (figura 2). Prima di essere attivato di nuovo occorrerà che tutti gli altri task ready siano stati eseguiti. Questo algoritmo dà a tutti i task uguale opportunità di essere eseguiti, ma non offre garanzie sui tempi necessari per eseguire un "giro" completo, e quindi su quanto tempo sarà necessario prima che un determinato task sia eseguito di nuovo. Dal momento che il passaggio da un task ad un altro avviene via software, questo tipo di scheduling può essere utilizzato anche senza fare ricorso alle interruzioni.
Un approccio diverso è quello di uno scheduling di tipo **preemptive** (con prelazione), in questo caso lo scheduler può interrompere forzatamente l'esecuzione di un task per eseguirne un altro, quando si verificano le condizioni opportune.

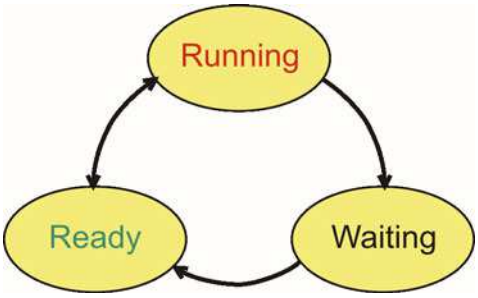


Figura 1 *Stati di esecuzione dei task e loro transizioni*

Normalmente lo scheduling preemptive è associato all'uso di **priorità** tra i task: se in seguito ad un evento un task a priorità alta passa dallo stato waiting a ready, ed in quel momento è eseguito un task a priorità più bassa, allora questo viene sospeso (cioè passerà da running a ready) in favore del primo (figura 3). Quando il task a priorità più alta tornerà allo stato waiting allora l'esecuzione del secondo potrà riprendere (sempre che non ci siano altri task a priorità più alta ready). Se tutti i task hanno priorità uguale si ha un round robin semplice. Ovviamente il multitasking preemptive offre maggiori garanzie nei confronti di vincoli temporali, e tra l'altro si presta bene ad essere associato al meccanismo delle interruzioni. Nei multitasking preemptive infatti lo scheduler verifica ad intervalli di tempo regolari, chiamati "**clock tick**" se lo stato dei vari task è cambiato. I clock tick (che sono relativamente frequenti, ed in genere sono generati dall'interrupt fornito da un timer hardware) servono anche per aggiornare variabili di stato, contatori e temporizzatori software.
Se ad un miltitasking preemtive descritto prima si aggiunge anche la caratteristica che tutti i task possono essere eseguiti per un intervallo di tempo fisso, dopo il quale si ha un cambio automatico di task, si ha un semplice sistema **operativo real-time** (RTOS). La caratteristica principale di un RTOS è che i tempi di esecuzioni e risposta sono deterministici, o comunque conoscibili con precisione e garantiti, almeno nel caso peggiore.

Sincronizzazione

I vari task possono essere dei programmi completamente indipendenti, tuttavia in molti casi alcuni di essi devono necessariamente collaborare e coordinarsi in qualche modo. Questo può accadere sia quando diversi task sono utilizzati per gestire un unico compito complesso, sia nel caso in cui più task devono utilizzare un'unica risorsa condivisa (una periferica, un'area di memoria, una porta di I/O…). I sistemi operativi mettono a disposizione diversi meccanismi per gestire la coordinazione e la sincronizzazione tra i task, i più comuni sono i **mutex**, i **semafori**, e le **code**.
I *mutex* (contrazione di "*mutual exclusion*", cioè "mutua esclusione") vengono utilizzati per condividere una risorsa tra due task. Essi funzionano più o meno come un microfono conteso da due presentatori: solo chi ha il microfono in mano può utilizzare l'amplificazione (o la linea). Nello stesso modo, solo uno dei due task può "prendere" il mutex, l'altro dovrà aspettare che questo venga rilasciato per poterlo prendere a sua volta. Il mutex è quindi una specie di flag binario, a cui il programmatore può assegnare un significato arbitrario. Ad esempio, se nel nostro sistema è presente una stampante, e ci sono due task ad un certo punto vogliono utilizzarla, occorrerà fare in modo che uno dei task eviti di mandarle dei caratteri prima che l'altro abbia terminato, altrimenti si otterrà una stampa contenente le due informazioni mischiate! Per fare questo si può creare un mutex, che sarà "preso" dal primo task che dovrà stampare. L'altro task vedrà che il mutex è impegnato, e si metterà in attesa fino a quando questo il mutex non si liberà. Sarà compito del primo task "rilasciare" il mutex quando avrà finito di stampare.
Se i task da coordinare sono più di due si possono utilizzare i **semafori**. Il loro funzionamento è identico a quello dei mutex, cioè solo uno dei task tra quelli che fanno riferimento allo stesso semaforo potrà aggiudicarsi il "possesso" del semaforo stesso. Il meccanismo è talmente simile a quello dei mutex, che è possibile anche utilizzare questi se il sistema operativo non dispone di semafori (occorrerà solo usarne più mutex, e gestire il loro stato in maniera coerente).

Figura 2 Esempio di scheduling cooperativo

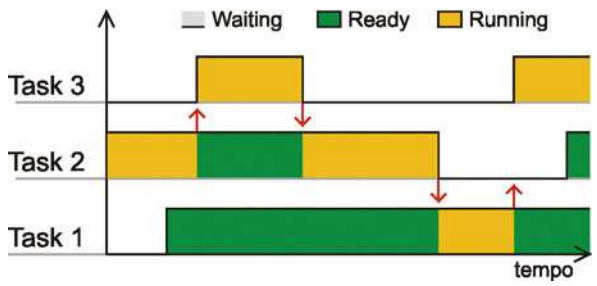


Figura 3 Esempio di scheduling preemptive (i task sono ordinati per priorità crescente)

RENDI IL TUO NATALE PIÙ LUMINOSO!

Tutti i prezzi si intendono IVA inclusa.

Babbo Natale su corda luminosa

Lunghezza cavo luminoso: 1m (cordone di alimentazione 2m); altezza di ciascun Babbo Natale (3 elementi): 40cm.

36,00 XMT120 - 120cm

13,50 XMSC3

13,00 XMT60 - 60cm

6,50 XMT25 - 25cm

Campana animata

Campana in movimento con 83 led. Alimentazione: adattatore 9 - 12 VDC o batteria alcalina 9V (non inclusa).

14,00 MK122 - kit

Candela elettronica in kit

5,50 MK167 - kit

Può funzionare in tre modalità: ad intensità di luce fissa, ad intensità di luce variabile, ad accensione/spegnimento con intensità di luce che cambia progressivamente. Alimentazione mediante batteria (non inclusa).

Babbo Natale con slitta

126 led multicolore animano questo gadget. Alimentabile a 12V per utilizzo in auto oppure con batteria alcalina 9V non compresa. Disponibile anche montato e collaudato.

21,00 MK116 - kit

33,00 MMK116 - montato

Candela elettronica

Simula la luce di una candela creando la stessa atmosfera. Completa di contenitore in vetro opaco. Alimentazione: 2 pile ministilo (non comprese).

2,50 XCML1

Albero di Natale a 134 led

Le 18 candele lampeggiano in modo casuale. Funzionamento a batteria o con alimentazione esterna a 12V per utilizzo in auto. Disponibile anche nella versione montata e collaudata.

20,50 MK117 - kit

32,50 MMK117 - montato

9,00 XMSC1

Babbo Natale su scala luminosa

Altezza Babbo Natale: 55cm; altezza scala: 70 cm (cavo di alimentazione 2m).

16,00 XMSC2

Stella lampeggiante a 35 led

Alimentazione: 1x9V (batterie non inclusa); dimensioni: 90 x 90mm. Disponibile in due versioni.

8,50 MK169Y - kit (led gialli)

MK169R - kit (led rossi)

Babbo Natale scalatore

Lunghezza corda: 2m; altezza di ciascun Babbo Natale (3 elementi): 40 cm.

Abete a fibre ottiche

La rotazione di un disco colorato, genera, attraverso le fibre ottiche, un bellissimo gioco di luci multicolore. Alimentazione: 6Vac/2A (adattatore di rete compreso). Disponibile in 3 differenti dimensioni (25/60/120cm).

Stella con 60 led multieffetto

Possibilità di creare 24 effetti luminosi. Alimentazione mediante adattatore da rete o a batterie 9~12 Vdc (non inclusi); dimensioni: 105 x 110 x 35 mm.

17,00 MK170 - kit

Albero di Natale in 3D

Composto da 16 led lampeggianti colorati. Funziona a batteria a 9V non inclusa oppure in auto a 12V. Basso consumo.

12,50 MK130 - kit

Cavo multicolore con controller

Una serie di lampadine colorate si accendono comandate da un controller in grado di eseguire 8 differenti sequenze. Per uso interno. Alimentazione 230 Vac. Lunghezza: 8m.

21,00 RL284C

Cavo multicolore 160 lampadine

Lunghezza: 10m. Comprende anche un controller in grado di eseguire 8 sequenze di accensione. Alimentazione: 24V (adattatore 230Vac incluso). Disponibile in due versioni.

16,00 XML8 - colorato

16,00 XML8W - bianco

Albero di Natale SMD

8,00 MK142 - kit

Mini albero con 6 led lampeggianti in sequenza. E' alimentato con un elemento al Litio da 3V. Basso consumo.

Albero di Natale a 16 led

Funziona con batteria a 9V non inclusa. Disponibile sia nella versione in scatola di montaggio che già montato e collaudato.

10,50 MK100 - kit

12,50 MMK100 - montato

Cavo luminoso colorato con led (l=5m)

23,00 RLL15R

54,00 RLL15G

24,00 RLL15Y

33,00 RLL15B

52,00 RLL15C

Cavo composto da una serie di led colorati che garantiscono un consumo bassissimo (solo 4,4W per metro!) ed una durata praticamente illimitata (25000h!). Diametro cavo: 13mm. Disponibile in differenti colori.

Cavo luminoso colorato 9m

15,50

RL19C - trasparente

RL19B - blu

RL19G - verde

RL19R - rosso

RL19Y - giallo

A tenuta stagna, è ideale per innumerevoli applicazioni. L unità viene fornita pronta all uso completa di un cordone di alimentazione. Disponibile in differenti colorazioni.

Cavo luminoso colorato 45m

Sistema a tenuta stagna utilizzabile in numerose applicazioni. Funzionamento affidabile e sicuro, vita di ogni lampadina: 20000h! Disponibile in differenti colorazioni.

56,00

RL145B - blu

RL145G - verde

RL145Y - giallo

FUTURA ELETTRONICA

Via Adige, 11 - 21013 Gallarate (VA)
Tel. 0331/799775 - Fax. 0331/778112
www.futuranet.it

Disponibili presso i migliori negozi di elettronica o nel nostro punto vendita di Gallarate. Caratteristiche tecniche e vendita on-line: www.futuranet.it

Un altro meccanismo molto utile alla sincronizzazione e comunicazione tra i task è quello delle **code** (dette anche "*message queues*"). Le code vengono utilizzate per passare dei messaggi (o dati) da un task ad un altro. Ad esempio il task che gestisce un display potrebbe ricevere il testo da visualizzare dal task che gestisce la comunicazione seriale via RS232. Il task del display in questo modo potrebbe rimanere in attesa (waiting) fino a quando non arriva un messaggio nella coda, una volta utilizzato il messaggio la coda sarebbe di nuovo vuota, ed il task andrebbe di nuovo nello stato waiting.
Le code comunque possono essere utilizzate in modo del tutto arbitrario, è possibile quindi usarle per implementare anche meccanismi più complessi. Se il sistema operativo non supporta le code è sempre possibile rimediare utilizzando un'area di memoria condivisa ed qualche mutex, o viceversa.

Effetti indesiderati

Il fatto di eseguire diversi task "contemporaneamente", può dar luogo ad una serie di problemi non sempre facilmente intuibili o prevedibili. Alcuni di questi sono simili a quelli visti a proposito delle interruzioni. Ad esempio il problema delle **sezioni critiche** esiste anche in un ambiente multitasking (in particolare se di tipo preemptive), in quanto l'esecuzione di un task può essere interrotta in favore di un altro, ed eventuali dati o variabili potrebbero essere scritte in maniera incompleta. Per risolvere questo problema diversi OS mettono a disposizione delle particolari funzioni per delimitare le sezioni critiche. Anche il problema del salvataggio e del cambio del contesto (context switch) da un task ad un altro è molto sentito nel caso degli OS, in quanto esso comporta una perdita di tempo notevole, che va ad incidere alla fine sul "tempo di risposta" dei programmi. Per limitare questo effetto l'unico accorgimento possibile è accorpare più task in uno solo.
Altri effetti indesiderati nascono dall'interazione tra i task (e quindi dall'uso di mutex, semafori e strumenti simili). Il più noto di questi effetti è il **deadlock**, che si verifica quando più task attendono reciprocamente che gli altri compiano una certa azione: alla fine tutti i task rimangono bloccati in una situazione senza uscita. É un po' come se due lottatori di Kick Boxing restassero fermi aspettando entrambi che sia l'altro a fare la prima mossa... L'unico modo per uscire da un deadlock spesso è riavviare il sistema! Invece l'unico modo per evitare i deadlock è riflettere bene in fase di progetto sui metodi di sincronizzazione che si vogliono utilizzare: alcuni potrebbero portare a deadlock, altri sicuramente no, per altri ancora potrebbe essere difficile prevederlo.

Uso dei sistemi operativi

Per potere utilizzare le funzioni di un sistema operativo è necessario prima di tutto strutturare e scrivere il programma in modo opportuno, secondariamente è necessario "linkare" il nostro codice al sistema operativo stesso. In genere per utilizzare le funzioni del sistema operativo è necessario includere nel nostro codice delle apposite librerie che permettono di richiamare le funzioni. Il risultato è in alcuni casi un programma indipendente che può essere eseguito dal sistema operativo quando richiesto, in altri casi invece il programma ed il sistema operativo vengono compilati assieme per formare un unico eseguibile che contiene entrambi (questo approccio è più comune nel campo dei sistemi embedded). In questo caso si parla di kernel o micro-kernel più che di OS. La scrittura del codice è identica nei due casi: occorre soltanto includere gli header del sistema operativo, e strutturare il programma in modo opportuno. Come va scritto un programma quando si utilizza un OS multitasking? La sua struttura è abbastanza diversa da quella di un programma tradizionale. In quest'ultimo caso infatti si ha una funzione main che gestisce il flusso di esecuzione principale, e tante altre funzioni che vengono richiamate da questa. Il tutto insomma viene eseguito in maniera strettamente sequenziale (se si escludono le interruzioni). Se il nostro programma vuole sfruttare le caratteristiche di multitasking dell'OS, la sua struttura cambia radicalmente: nel main verranno creati ed inizializzati i vari task e gli strumenti ausiliari quali mutex, semafori e code. I task verranno poi avviati, e saranno quindi eseguiti in concorrenza dallo scheduler. Da questo punto l'esecuzione del programma si divide in più linee di esecuzione diverse, tante quanti sono i task. Per mostrare quanto detto considereremo di seguito un esempio reale: FreeRTOS. FreeRTOS è un semplice ma efficiente kernel real time scritto in ANSI C (con qualche piccola parte in assembly), dotato di tutte le caratteristiche viste prima, e disponibile gratuitamente come open source su Internet. FreeRTOS è stato

pensato per funzionare su piccoli sistemi a microprocessore o microcontrollore, e quindi richiede risorse minime per poter essere utilizzato (al contrario di altri OS più noti come uCLinux, eCos, o QNX). É disponibile per molti processori, tra cui: ARM7, AVR ATMega, PIC 18x, H8, 8051, e tanti altri. Maggiori informazioni ed i sorgenti possono essere trovati sul sito www.freertos.org.

Creare, eseguire e distruggere i task

Un task non è altro che una normale funzione che contiene un loop infinito (in modo simile a quanto di solito avviene nel main). Per creare un task è quindi necessario prima di tutto scrivere la funzione che lo costituisce, e poi assegnarla ad un task utilizzando la funzione *xTaskCreate* di FreeRTOS. Similmente per distruggerlo (quando e se sarà necessario), si potrà utilizzare la funzione *vTaskDelete*:

```
#include "FreeRTOS.h"
#include "task.h"

// Funzione del primo task
void MioTask(void *parametri)
{
  while(1)
  {
    // Codice del task
  }
}

// Funzione del secondo task
void AltroTask(void *parametri)
{
  while(1)
  {
    // Codice del task
  }
}


main()
{
  int parametro=2;
  xTaskHandle MioHandle,
              AltroHandle;

  // Creazione dei task
  xTaskCreate(MioTask,
              "PRIMO",
              STACK_SIZE,
              &parametro,
              5,
              &MioHandle);

  xTaskCreate(AltroTask,
              "SECONDO",
              STACK_SIZE,
              &parametro,
              2,
              &AltroHandle);

…
  // Avvia esecuzione tasks
  vTaskStartScheduler();
…

  // Distruzione dei tasks.
  vTaskDelete(MioHandle);
  vTaskDelete(AltroHandle);
…
}
```

Per utilizzare FreeRTOS occorre includere i due file header relativi. La funzione xTaskCreate prende come parametri il nome della funzione del task (che, ricordiamo, è in fin dei conti un puntatore), il cui prototipo deve essere quello mostrato, una stringa arbitraria che costituisce il nome del task, la dimensione dello stack (cioè il numero di variabili che utilizza la funzione), un puntatore ad una variabile che contiene i parametri da passare alla funzione (che possono essere di qualsiasi tipo, anche strutturato), la priorità del task, che per default è un numero compreso tra 0 e 5 (essendo 5 la massima), ed infine un puntatore ad una variabile in cui verrà restituito l'handle al task creato. Un handle non è altro che una specie di identificativo, che serve per riferirsi al task creato (un po' come avviene con i file). Anche se non è mostrato, la funzione restituisce un valore per comunicare se la creazione è andata a buon fine. É possibile creare diversi task in questo modo. Quando saranno stati tutti creati, sarà possibile avviare la loro esecuzione, invocando la funzione vTaskStartScheduler. A questo punto il controllo del programma passa allo scheduler che gestirà l'esecuzione dei vari task secondo le sue politiche. La funzione vTaskStartScheduler non terminerà fino a quando qualcuno dei task eseguiti

non invocherà la funzione vTaskEndScheduler. Solo in questo caso la funzione terminerà ed il rimanente codice del main riprenderà ad essere eseguito. Si arriverà in questo modo alla funzione vTaskDelete che provvederà a distruggere i task.
Un task può essere creato anche all'interno di un altro task, e può anche auto-distruggersi, secondo le esigenze del programma, non è necessario che queste funzioni siano svolte nel main.

Code

FreeRTOS mette a disposizione delle code, che sono internamente utilizzate anche per implementare i mutex ed I semafori. É possibile creare e cancellare delle code con le seguanti funzioni:

```
xQueueHandle MiaCoda;
MiaCoda=xQueueCreate(10, sizeof(long));
...
vQueueDelete(MiaCoda);
```

In questo modo abbiamo creato una coda di 10 elemnti, ciascuno grande 4 byte (dimenzione di una variabile di tipo long). La funzione restituisce un handle alla coda creata, o un codice di errore. Per cancellare la coda è sufficiente passare l'handle alla funzione vQueueDelete. Per inviare un messaggio ad una coda si può scrivere:

```
r=xQueueSend(MiaCoda, &datoTX,
(portTickType) 10);
```

La funzione prende in ingresso l'handle della coda, il puntatore al dato da inviare, ed il tempo (espresso in tick) che vogliamo aspettare in caso la coda sia piena. In questo tempo il task passerà allo stato waiting, e ne uscirà o allo scadere del tempo, o quando nella coda si sarà creato spazio. La funzione restituisce un valore che indica l'esito positivo o negativo (coda piena) dell'operazione. La funzione duale di quella appena vista è la seguente:

```
r=xQueueReceive(MiaCoda, &datoRX,
(portTickType) 20);
```

Un task può usare questa funzione per leggere un dato dalla coda, o per attenderlo se non è disponibile. Anche in questo caso il task verrà sospeso in attesa dell'esito dell'operazione.

Mutex e semafori

FreeRTOS utilizza internamente il meccanismo delle code per gestire i semafori. In realtà comunque si tratta di semafori binari, quindi di mutex. Per ottenere dei semafori veri e propri è necessario utilizzare più mutex, come spiegato prima. Un semaforo binario può essere creato con la seguente funzione:

```
xSemaphoreHandle xSemaforo;
vSemaphoreCreateBinary(xSemaforo);
```

Per "prendere" un semaforo si utilizza la seguente funzione:

```
r=xSemaphoreTake(xSemaforo, (portTickType) 30);
```

La funzione specifica, non solo di prendere il semaforo (se è disponibile), ma anche di aspettare per un certo periodo di tempo (30 tick) che il semaforo si liberi, prima di concludere che è occupato. Il valore restituito sarà la costante pdTRUE, se il semaforo si è liberato, ed è stato preso.
Una volta completata l'operazione sulla risorsa condivisa, il task dovrà rilasciare il semaforo utilizzando la seguente funzione:

```
xSemaphoreGive(xSemaforo);
```

CONCLUSIONE

Con le poche e semplici funzioni descritte è possibile scrivere programmi multitasking, che impiegano anche complessi meccanismi di sincronizzazione.
Sarà possibile in questo modo non solo rendere più efficienti e modulari i propri programmi, ma anche soddisfare vincoli temporali abbastanza critici, sfruttando le caratteristiche real time di FreeRTOS.
Probabilmente apparirà ora più chiaro quali vantaggi può offrire l'uso di un OS. Tuttavia nonostante questo, la scelta di scrivere un programma stand-alone piuttosto che ricorrere ad un OS è lasciata all'esperienza del programmatore: occorre sempre bilanciare bene vantaggi e svantaggi offerti da ciascun approccio.

Si conclude con questa puntata la serie Vitamina C. Ringrazio tutti per l'attenzione dimostrata e per il positivo feedback.

Conoscere ed usare

INWARE EDIZIONI

"Conoscere ed usare" è la nuova collana di libri edita da Inware Edizioni, dedicati a chi intende utilizzare dispositivi e componenti elettronici di nuova concezione, per conoscerli ed usarli nel modo più semplice e veloce possibile mediante numerosi esempi pratici.

Display LCD

Una guida all'utilizzo dei moduli alfanumerici basati sul controller HD44780, moduli grafici con controller KS0108 e non solo. Il testo tratta anche i display LED a sette segmenti e i display LCD passivi. Numerosi gli esempi pratici di impiego dei vari dispositivi: dal contatore a 7 segmenti al termometro LCD fino al pilotaggio dei moduli alfanumerici mediante PICmicro e PC.

COD. FE-06
€ 16,50

PICmicro™

Per conoscere a fondo i PICmicro seguendo un percorso estremamente pratico e stimolante. Vengono analizzate la struttura interna, le porte di I/O, le tecniche di uso del Watchdog Timer, la gestione della EEPROM interna e molti altri argomenti attraverso montaggi pratici e semplici da realizzare. Il testo descrive l'uso di MPLAB®, l'ambiente di sviluppo Microchip per la gestione dei progetti basati su PICmicro e descrive, in maniera approfondita, tutte le istruzioni assembler e molte delle direttive del compilatore. Al testo è allegato un utilissimo CDROM che, oltre ai sorgenti e gli schemi dei progetti presentati nel testo, contiene moltissimi programmi di utilità e molta documentazione.

COD. FE-18
€ 29,00 (contiene CD-ROM)

Acquista direttamente sul sito
www.farelettronica.com
o telefona al numero
02.66504794

Riflettori su...

EIKON

Dotare di comunicazione wireless di tipo standard i sistemi embedded apre nuove opportunità per i prodotti esistenti o di nuova concezione. In questo modo i moderni smartphones e palmari possono diventare l'interfaccia utente dei vostri sistemi elettronici a microprocessore espandendone il campo di impiego ed aumentandone il valore percepito dall'utente.

Eikon progetta e produce moduli e prodotti basati su protocolli wireless standard mondiali quali Bluetooth e Wi-Fi. Le nostre soluzioni sono di tipo "cable replacement" perchè non necessitano di modifiche all'hardware o al firmware dei vostri sistemi embedded in quanto tutto il software necessario gira a bordo del nostro hardware. L'esperienza sviluppata con le oltre 200 applicazioni consegnate in tutto il mondo, la velocità nelle consegne anche di prodotti OEM grazie alla produzione interna e il supporto tecnico in real time anche su internet ci rendono un'azienda moderna e con visibilità globale.

ELIMINAZIONE CAVI SERIALI

Eliminare i cavi seriali (Uart, RS232 o RS485) è possibile sia con Bluetooth che con Wi-Fi. Bluetooth ha il vantaggio di essere integrato sempre di più nei palmari e smartphones, mentre Wi-Fi supera i limiti di Bluetooth nelle distanze e per l'immediata connessione in rete tramite Access point. Le nostre soluzioni sono tutte scalabili in quanto possiamo partire dai moduli fino ai prodotti finiti senza variazioni sui firmware che possono essere portati da un prodotto all'altro senza costi aggiuntivi.
Questo consente ai nostri clienti di iniziare con soluzioni "end user" per poi passare ai moduli SMT più economici ed adatti a produzioni in volumi medio-alti.

Bluetooth

UARTDNG105 è il nostro modulo migliore per eliminare i cavi nelle comunicazioni seriali. E' dotato di antenna integrata, alimentazione 5V o 3,3V e Serial port profile. Una volta alimentato e collegato alla seriale (TX,RX ed RTS/CTS se utilizzati) del vostro sistema sarete immediatamente in grado di scambiare dati sulla COM virtuale Bluetooth con un palmare o un qualsiasi altro dispositivo wireless dotato di Serial port profile.

Il prodotto viene consegnato configurato per l'utente oppure è possibile acquistare la DEVBOARD001 per settarne i parametri di funzionamento. Un'alternativa più economica è UARTDNG101 dotato di porta UART (a livello 3,3V e TTL). Il firmware ed il funzionamento sono identici a quelli dell'UARTDNG105, cambiano solo le dimensioni ed il pinout.
A livello di prodotto finito la soluzione ideale da posizionare all'esterno dell'apparecchiatura è RS232CL1E05: adattatore seriale Bluetooth in classe 1 (100m). Il prodotto viene consegnato

Figura 1 UARTDNG105)

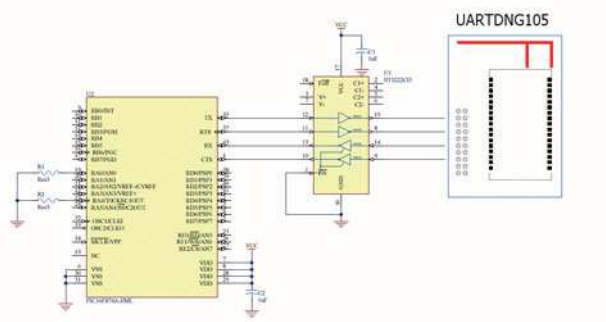


Figura 2 Schema embedded con UARTDNG105

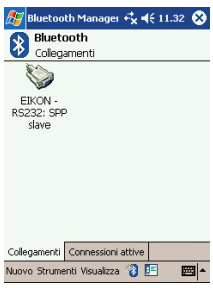


Figura 3 Connessione Bluetooth con PDA

“Wireless Made Easy”

completo di alimentatore e software per la configurazione dei suoi parametri da PC. Una volta configurato anche questo adattatore non necessita di drivers per il funzionamento e può essere alimentato anche tramite il pin 9 del connettore DB9 della seriale oppure su richiesta lo possiamo fornire con portabatteria (4 stilo AA).

Wi-Fi

Per collegare ad una rete locale un dispositivo seriale è possibile utilizzare MULTIWI001 : convertitore da RS232 a Wi-Fi 802.11b completamente configurabile. Una volta settato il prodotto tramite il suo software fornito in dotazione esso si connette automaticamente all’access point della rete Wi-Fi e il vostro sistema embedded viene quindi dotato di un proprio indirizzo IP consentendovi di controllarlo tramite Lan o Internet. Sono numerosi i software gratuiti su Internet per creare delle COM virtuali over IP che rendono molto semplice questo tipo di controllo remoto.

APPLICAZIONI AUDIO BLUETOOTH

Il nuovo modulo UARTDNG300 è caratterizzato da dimensioni estremamente contenute e antenna SMT (o esterna con connettore SMA) e consente di scambiare audio analogico con palmari, smartphones o auricolari Bluetooth.

Il nostro firmware Audio Headset consente di connettere il modulo a telefonini o palmari che immediatamente riconoscono il sistema come un normale auricolare Bluetooth dando quindi la possibilità di creare soluzioni viva voce per auto, moto, nautica ecc.

Al contrario, quando viene caricato il nostro firmware Audio Gateway UARTDNG300 si connette automaticamente ad auricolari standard Bluetooth. In questo modo i nostri clienti hanno realizzato sistemi a comando vocale molto eleganti e funzionali in quanto l’utente finale può utilizzare un auricolare Bluetooth qualsiasi tra quelli disponibili sul mercato.

Aprire i vostri sistemi al mondo wireless Audio Bluetooth vi consentirà sicuramente di creare nuove applicazioni dall’altissimo valore percepito dagli utenti che sono oggi sempre più abituati ad utilizzare auricolari Bluetooth e telefonini.

OEM E CUSTOM

Siamo in grado di sviluppare hardware e firmware custom per realizzare soluzioni ad hoc per i nostri clienti. I nostri prodotti sono dotati di pin di I/O che possiamo utilizzare ad esempio per monitorare lo stato della connessione o l’entrata in pairing del sistema.

Abbiamo anche realizzato sistemi completamente gestiti dai nostri moduli senza bisogno della presenza di un microprocessore per ottimizzare spazi e costi (Gps, lettori barcode, sistemi di lettura/scrittura RFID ecc.)

Figura 4 Convertitore Seriale-Wifi

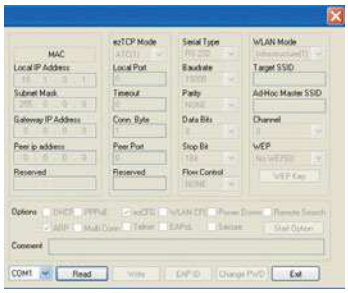


Figura 5 Software configurazione

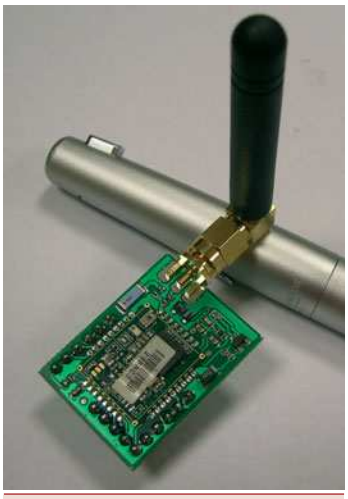


Figura 6 UARTDNG300

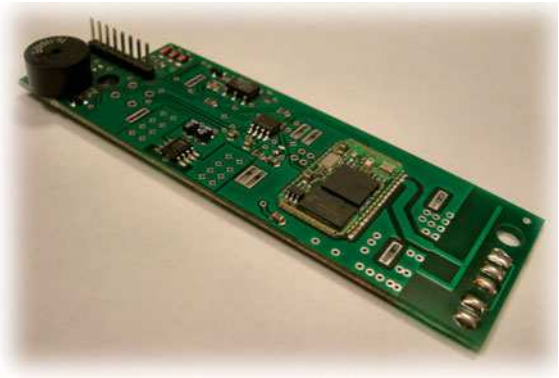


Figura 7 Foto OEM

Avere la linea di produzione elettronica interna ci consente di passare dal progetto al prodotto in meno di 20gg tenendo sotto controllo costante il livello qualitativo dei materiali, dell'assemblaggio e della fase di test e programmazione.

IL WIRELESS DEL FUTURO: ZIGBEE E RETI MESH

Superare i limiti delle attuali tecnologie esistenti (consumi, costi, tempistiche di accesso, tipologie di rete ecc.) è il motivo della nascita dello standard wireless Zigbee. Zigbee sarà leader nelle applicazioni di Domotica e sensoristica grazie ai consumi ridottissimi (sempre più dispositivi saranno alimentati a batteria) e alla nuova tipologia di network che consente di realizzare (Mesh network). In tal modo i dispositivi potranno comunicare tra loro non in modo diretto ma rimbalzando all'interno della Mesh (un po' come succede su Internet) in modo da avere un vero risparmio energetico sulla parte RF.

Figura 8 Zigbee

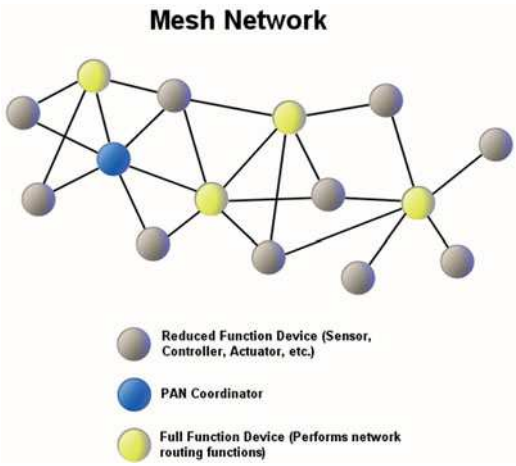


Figura 9 Rete Mesh

Eikon lancerà i propri moduli e prodotti Zigbee ad inizio 2006, ma come anticipazione possiamo dire che saranno basati su un set di comandi AT e saranno dotati di interfaccia hardware UART, RS232 e USB.

LE NUOVE FRONTIERE EIKON: SCHEDE CPU SCALABILI PER APPLICAZIONI AVANZATE

Il 2006 non sarà per la nostra azienda solo l'anno dello Zigbee, ma anche quello di una nuova linea di prodotti, non wireless, ma di mobile computing. Abbiamo deciso di creare una linea di schede CPU che supportino i nuovi sistemi operativi Microsoft Embedded (Windows CE ad esempio) e Linux.

L'esigenza è nata per risolvere delle specifiche esigenze dei nostri clienti che oggi vogliono andare sempre più velocemente sul mercato con prodotti sempre più evoluti (ad esempio con LCD touch screen come interfaccia) non investendo sull'hardware in quanto i loro volumi non lo consentirebbero.

Raccogliendo tutte queste esigenze stiamo lavorando alla nuova scheda CPU che seguirà la nostra filosofia del "made easy": il cliente potrà direttamente sviluppare i propri applicativi software e caricarli sulla nostra scheda che si preoccuperà della gestione dell'hardware (RS232, USB, Ethernet, Audio, VGA, IO analogici e digitali ecc).

Le nostre schede CPU saranno scalabili ovvero saremo in grado di fornire soluzioni con costi e caratteristiche diverse, ma basate sulla stessa piattaforma hardware in modo da dare al cliente la flessibilità di creare una linea di prodotti diversi solo per interfacce e quantità di memoria. Tramite questi cores si potranno creare dispositivi palmari, automotive, controlli industriali, mobile entertainment.

Per informazioni potete far riferimento a:

Eikon srl
Tel +39 0721 877365
Fax +39 0721 897679
info@eikonsite.it
www.eikonsite.it

Le so tutte!!!

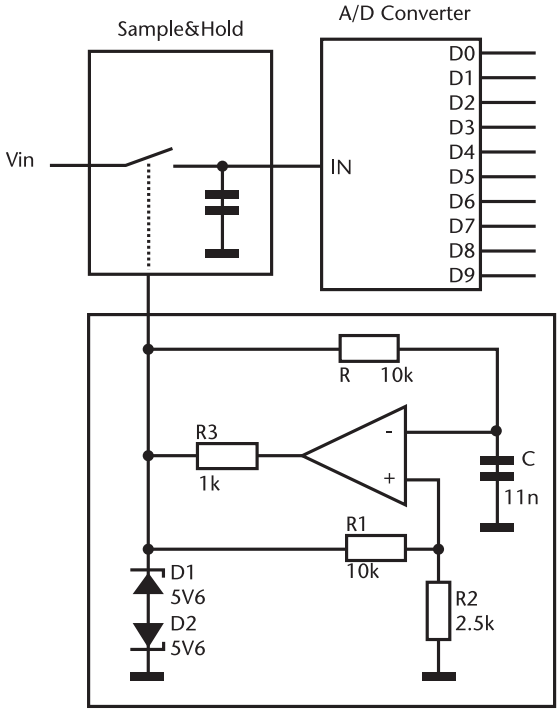
Metti alla prova le tue conoscenze e le tue capacità rispondendo al quesito, se la risposta si rivelerà esatta potrai vincere l'esclusiva felpa pile di Fare Elettronica.

Ogni mese sulle pagine della rivista troverai la soluzione del quesito del mese precedente e il nome del vincitore di uno dei quesiti pubblicati. Tutti i partecipanti riceveranno un coupon del 10%* di sconto utilizzabile per un acquisto nello shop di www.farelettronica.com.

* Validità 3 mesi

Quesito - LST24610

Per la registrazione digitale di un segnale audio di qualità telefonica (banda di circa 5KHz) viene utilizzato un modulo sample&hold per il campionamento del segnale ed un convertitore Analogico/Digitale con risoluzione di 10bit secondo lo schema a blocchi di figura. Il modulo sample&hold è pilotato da un multivibratore astabile che genera un'onda quadra la cui frequenza determina la frequenza di campionamento del segnale analogico di ingresso. Con i valori riportati, il periodo di campionamento è di circa 90µs. Volendo utilizzare lo stesso sistema per digitalizzare un segnale audio di qualità superiore avente una banda di 10KHz, quali modifiche dovremmo applicare al circuito?



Scadenza: il termine ultimo per rispondere è il 31 Dicembre 2005

Regolamento

1) Il quiz è aperto a tutti i lettori.

2) Saranno considerate esclusivamente le risposte pervenute entro la scadenza indicata nel quesito.

3) Inviare la risposta compilando il modulo su **www.farelettronica.com/lesotutte**.

Sì approfitto della fantastica occasione e mi abbono a 11 numeri di Fare Elettronica e pagherò solo € 45,00

FE - 246

DATI PERSONALI

Nome .....

Cognome .....

Via ..... n° ....

Cap ..... Città ..... Prov ....

Tel ..... Fax .....

Email .....

Ragione Sociale .....

P.Iva ..... ❍ Ricevuta

Privacy. Ai sensi del Decr. Lgs. 196/2003 la informiamo che i dati trasmessi verranno impiegati coi principali scopi di indagini di mercato e nelle modalità previste dallo stesso, prevalentemente con mezzi informatici. Il conferimento, di norma facoltativo, è obbligatorio per permettere il rapporto commerciale. È in ogni caso fatto diritto dell'interessato esercitare i propri diritti, nei modi previsti dal "Titolo II art. 7" della legge sopra citata, scrivendo a Inware Edizioni Via Cadorna 27 – 20032 Cormano o trmite email a info@inwaredizioni.it

ABBONATI SUBITO

Compila, ritaglia e spedisci via fax questo coupon allo **02-66508225**

Spedisci questa pagina in busta chiusa a:
INWARE Edizioni
Via Cadorna, 27/31 - 20032 Cormano (MI)

Chiamaci allo **02-66504794**

Abbonati on-line sul sito
www.farelettronica.com/abbonamento

MODALITÀ DI PAGAMENTO

❍ **CARTA DI CREDITO**

❑ American Express ❑ Visa ❑ Mastercard

Titolare .....

n° ..... scad .....

❍ **VERSAMENTO SUL CCP 22790232**

Allegare la ricevuta (o copia) del versamento intestato a Inware Srl, indicando nella causale: "Abbonamento Fare Elettronica"

❍ **BONIFICO BANCARIO**

Appoggiarlo su: **Poste Italiane - CIN: Z - ABI: 07601 CAB: 01600 - C/C: 000022790232** intestato ad **Inware srl**

❍ **ALLEGO UN ASSEGNO**

intestato a Inware Srl

Firma .....

Abbonati subito!

- ✔ **Risparmierai ben 15,50 euro sul prezzo di copertina**
- ✔ **Avrai la garanzia del prezzo bloccato per un anno**
- ✔ **La rivista ti sarà recapitata comodamente a casa**
- ✔ **Compreso con l'abbonamento (o il rinnovo) riceverai un buono sconto del 20%* per un tuo prossimo acquisto sul sito www.farelettronica.com**
- ✔ **Compila il cedolino qui a fianco e invialo in busta chiusa o via fax, riceverai un gradito regalo a sorpresa**

* Validità 3 mesi

INWARE EDIZIONI